



HAL
open science

Contrôle du spectre électromagnétique d'un circuit numérique asynchrone

Sophie Germain

► **To cite this version:**

Sophie Germain. Contrôle du spectre électromagnétique d'un circuit numérique asynchrone. Micro et nanotechnologies/Microélectronique. Université Grenoble Alpes, 2019. Français. NNT : 2019GREAT053 . tel-02523608

HAL Id: tel-02523608

<https://theses.hal.science/tel-02523608>

Submitted on 29 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : Nano Electronique et Nano Technologies

Arrêté ministériel : 25 mai 2016

Présentée par

Sophie GERMAIN

Thèse dirigée par **Laurent FESQUET**
et co-encadrée par **Sylvain ENGELS**

Préparée au sein du **Laboratoire Techniques de l'Informatique et de la Microélectronique pour l'Architecture des systèmes intégrés** dans l'**École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Contrôle du spectre électromagnétique d'un circuit numérique asynchrone

Thèse soutenue publiquement le **25 novembre 2019**, devant le jury composé de :

Monsieur Luc HEBRARD

Professeur, Université de Strasbourg, Président

Madame Sonia BEN DHIA

Professeur, INSA Toulouse, Rapporteur

Monsieur Lionel TORRES

Professeur, Université de Montpellier, Rapporteur

Monsieur Fabien NDAGIJIMANA

Professeur, Université Grenoble Alpes, Examinateur

Monsieur Philippe GALY

Ingénieur, STMicroelectronics, Invité

Monsieur Laurent FESQUET

Maître de Conférences, Université Grenoble Alpes, Directeur de thèse

Monsieur Sylvain ENGELS

Ingénieur, STMicroelectronics, Co-Encadrant de thèse



Table des Matières

INTRODUCTION	1
CHAPITRE I. ETAT DE L'ART	5
I. 1. Introduction.....	6
I. 2. Les circuits asynchrones	6
I. 2. 1. Limitations des circuits synchrones	6
I. 2. 2. Conception des circuits asynchrones	7
I. 2. 2. 1. Principe	7
I. 2. 2. 2. Porte de Muller	8
I. 2. 3. Protocoles de communication	10
I. 2. 3. 1. Protocole deux phases.....	10
I. 2. 3. 2. Protocole quatre phases.....	11
I. 2. 4. Classes des circuits asynchrones.....	13
I. 2. 4. 1. Insensibles aux délais	13
I. 2. 4. 2. Quasi-Insensibles aux délais	13
I. 2. 4. 3. Indépendant de la vitesse.....	14
I. 2. 4. 4. Micropipeline.....	14
I. 2. 5. Avantages des circuits asynchrones.....	15
I. 2. 5. 1. Calcul en temps minimum	15
I. 2. 5. 2. Modularité	15
I. 2. 5. 3. Basse consommation	15
I. 2. 5. 4. Faible émission électromagnétique.....	16
I. 3. Rayonnement électromagnétique des circuits numériques	16
I. 3. 1. Champ électromagnétique	16
I. 3. 2. Interférences électromagnétiques	17
I. 3. 3. Compatibilité électromagnétique.....	18
I. 3. 4. Réduction du rayonnement dans les circuits synchrones.....	19
I. 3. 4. 1. Désactivation de l'horloge	19
I. 3. 4. 2. Génération d'horloge à spectre étalé.....	21
I. 3. 5. Réduction du rayonnement dans les circuits asynchrones.....	22
I. 4. Objectifs de la thèse.....	23
I. 5. Conclusion	24
CHAPITRE II. PRINCIPES DU CONTROLE DU RAYONNEMENT	27
II. 1. Introduction.....	28
II. 2. Premières tentatives de contrôle du spectre.....	28
II. 2. 1. Méthode synchrone.....	29
II. 2. 2. Méthode asynchrone.....	32
II. 3. Introduction d'un gabarit	34
II. 3. 1. Principe	34
II. 3. 2. Deux Exemples de Normes	35

II. 3. 2. 1.	Communication Ultra-Large Bande	35
II. 3. 2. 2.	Communication en Champ Proche	36
II. 4.	Présentation de notre méthode.....	37
II. 4. 1.	Hypothèse pour l'émission électromagnétique.....	37
II. 4. 2.	Ajout de délais	38
II. 4. 3.	Détermination des valeurs de délais	40
II. 4. 3. 1.	Approche analytique	41
II. 4. 3. 2.	Approche algorithmique.....	43
II. 5.	Conclusion	44
 CHAPITRE III. MODELISATION DE HAUT NIVEAU DU COURANT.....		47
III. 1.	Introduction.....	48
III. 2.	Réseau de Petri temporel.....	48
III. 2. 1.	Principe	48
III. 2. 2.	Utilisation.....	50
III. 2. 3.	Instanciation	51
III. 3.	Estimation du courant.....	54
III. 3. 1.	Modèle d'un pic de courant.....	55
III. 3. 2.	Modèle de la consommation de courant.....	59
III. 4.	Conclusion	60
 CHAPITRE IV. FLOT DE CONCEPTION.....		63
IV. 1.	Introduction.....	64
IV. 2.	Présentation du flot de conception.....	65
IV. 3.	Conception du circuit	66
IV. 4.	Synthèse logique.....	68
IV. 4. 1.	Etude du réseau de contrôle.....	68
IV. 4. 2.	Rupture des boucles combinatoires	70
IV. 4. 3.	Définition des horloges.....	72
IV. 5.	Analyse temporelle statique.....	74
IV. 5. 1.	Analyse des chemins	76
IV. 5. 2.	Récupération des délais	77
IV. 6.	Co-Simulation	79
IV. 6. 1.	Algorithme Génétique	79
IV. 6. 2.	Simulation Analogique Rapide	80
IV. 6. 3.	Obtention d'une solution.....	81
IV. 7.	Implémentation	83
IV. 7. 1.	Implémentation synchrone.....	83
IV. 7. 2.	Implémentation asynchrone.....	83
IV. 8.	Conclusion	84

CHAPITRE V. CIRCUITS DE TEST ET MESURES	87
V. 1. Introduction.....	88
V. 2. Paramètres pour aider à contrôler le rayonnement.....	89
V. 2. 1. Concevoir le circuit en asynchrone	89
V. 2. 2. Résolution des délais	91
V. 2. 3. Changement d'architecture	93
V. 3. Circuit de test « Pegase »	94
V. 3. 1. Architecture du circuit	95
V. 3. 2. Simulations	96
V. 3. 3. Réalisation du circuit.....	97
V. 4. Mesures.....	98
V. 4. 1. Temporelle	99
V. 4. 2. Courant	100
V. 4. 3. Electromagnétique.....	102
V. 4. 3. 1. Mini cage de Faraday	103
V. 4. 3. 2. Cellule TEM.....	104
V. 4. 3. 3. Scanner de surface	104
V. 5. Circuit de test « Hydra ».....	111
V. 5. 1. Architecture du circuit	112
V. 5. 2. Simulation	113
V. 5. 3. Réalisation du circuit.....	114
V. 5. 4. Simulation après implémentation	116
V. 6. Conclusion	117
CONCLUSION GENERALE	119
RÉFÉRENCES	123
PUBLICATIONS DE L'AUTEUR	127

Introduction

Lors de son fonctionnement, un circuit intégré émet un champ électromagnétique dans son environnement. Or, si ce rayonnement n'est pas contrôlé, il peut perturber le fonctionnement d'un autre circuit, voire le détruire dans le pire des cas. Ce phénomène est appelé « interférence électromagnétique ». Il est également connu au sein d'un circuit intégré lorsqu'il y a un transmetteur radiofréquence et un contrôleur numérique. En effet, la partie numérique émet un spectre électromagnétique important qui perturbe le fonctionnement de la partie analogique radiofréquence [1]. Pour éviter les interférences, des normes de Compatibilité ElectroMagnétique (CEM) ont été définies pour régir les émissions des circuits en fonction de leur application. Le respect de ces standards CEM a toujours été un enjeu important pour les concepteurs de circuits intégrés.

Il existe des techniques pour augmenter l'immunité des circuits en protégeant les parties sensibles avec un blindage [2]. Néanmoins, il n'existe pas à notre connaissance une technique permettant de contrôler de façon quantitative le rayonnement électromagnétique dès l'étape de conception. En général, le spectre est mesuré après la fabrication du circuit et s'il ne respecte pas les normes, des corrections sont effectuées en modifiant l'architecture par exemple. Cette méthode empirique ne permet pas de prévoir les résultats avant la mesure et est donc coûteuse en temps et en argent.

Le fonctionnement des circuits synchrones avec un signal global qui cadence tous les éléments de mémorisation génère de forts appels de courant sur l'alimentation. Cela produit un spectre électromagnétique important avec les harmoniques du signal d'horloge. Il existe des méthodes qui permettent de réduire de façon qualitative le rayonnement des circuits synchrones. Cependant, les harmoniques du signal d'horloge sont très compliquées à réduire. C'est pourquoi, les circuits asynchrones sont une alternative très intéressante pour la conception des circuits numériques. Leur fonctionnement avec des contrôleurs locaux qui communiquent avec des signaux, de requête et d'acquiescement, permet d'éviter la commutation simultanée de tous les éléments de mémorisation du circuit. Les appels de courant sont répartis non-uniformément dans le temps, ce qui réduit le spectre électromagnétique et évite par construction l'apparition d'harmoniques liées à un signal d'horloge.

Ce travail de thèse est une collaboration entre l'équipe *Conception de Dispositifs, Circuits et Systèmes Intégrés (CDSI)* du laboratoire TIMA et le *Groupe Microcontrôleur et Circuits Intégrés Numériques* de STMicroelectronics. Un des axes de travail de l'équipe du laboratoire est de développer des techniques de conception asynchrone ou d'en utiliser ses propriétés pour améliorer les circuits. L'équipe à STMicroelectronics réalise la conception de circuits intégrés qui peuvent être composés d'une partie numérique et d'une partie analogique radiofréquence. L'objectif de ce travail est de proposer une méthode de conception pour les circuits numériques asynchrones afin de contrôler leur rayonnement électromagnétique.

Le chapitre I de ce manuscrit introduit, dans un premier temps, les circuits asynchrones. Les limitations des circuits synchrones sont exposées. La présentation de la conception des circuits asynchrones est alors proposée au lecteur ainsi que les protocoles utilisés pour transmettre les données d'un étage à l'autre. Les différentes classes de circuits asynchrones sont ensuite présentées ainsi que leurs hypothèses de conception. Les avantages de cette méthode de conception sont explicités, notamment en ce qui concerne les émissions électromagnétiques. Dans un second temps, le rayonnement des circuits numériques et ses problématiques sont présentés. La notion de compatibilité électromagnétique est alors abordée. Des méthodes de réduction du rayonnement pour les circuits synchrones sont décrites et l'avantage naturel des circuits asynchrones est illustré avec des exemples. Enfin, l'objectif de ce travail de thèse est exposé.

Le principe du contrôle du rayonnement électromagnétique des circuits numériques est présenté dans le chapitre II. Les premières tentatives de contrôle du spectre sur des circuits synchrones et asynchrones sont décrites. La notion de gabarit fréquentiel, pour contrôler le rayonnement électromagnétique, est introduite. Ensuite, le lecteur prendra connaissance du principe de notre méthode pour contrôler le rayonnement avec l'utilisation du gabarit. Notre hypothèse de travail sur le lien entre la consommation de courant et le spectre du circuit est présentée. L'introduction de notre méthode pour déplacer les appels de courant en utilisant les propriétés des circuits asynchrones, plus particulièrement les circuits micropipelines, clôture ce chapitre.

Le chapitre III introduit une étape essentielle de notre méthode : la modélisation de haut niveau du courant. Notre simulateur analogique rapide est décrit. Il utilise un réseau de Petri temporel pour modéliser le circuit, le simuler et récupérer les instants de commutation des éléments de mémorisation. Ensuite, notre modèle de pic de courant est expliqué et appliqué pour chaque instant de mémorisation pour obtenir la consommation de courant du circuit.

L'utilisation de notre simulateur analogique rapide est exposée dans le chapitre IV, lors de l'explication de notre flot de conception pour contrôler le rayonnement électromagnétique. Les différentes étapes qui le composent sont expliquées à travers un circuit d'exemple. Elles s'assimilent aux étapes d'un flot de conception synchrone, la description au niveau registres, la synthèse logique, l'analyse temporelle statique et l'implémentation physique. Seule une étape de mise en forme du spectre a été ajoutée après l'analyse temporelle statique. Elle utilise un algorithme génétique et notre simulateur analogique rapide pour trouver les paramètres du circuit afin que son rayonnement respecte le gabarit fréquentiel. Toutes les étapes de notre flot de conception utilisent des outils standards.

Le dernier chapitre présente l'évaluation de notre méthode grâce à la présentation de deux circuits de test. Une première partie est consacrée aux paramètres que le concepteur peut changer dans son circuit pour faciliter l'étape de mise en forme du spectre. Ensuite, l'architecture et une simulation du fonctionnement ainsi que la réalisation du premier circuit

de test sont présentées. Celui-ci a été fabriqué en technologie CMOS 40 nm STMicroelectronics. Une partie est consacrée aux différentes mesures réalisées, dont des mesures électromagnétiques qui ont donné des résultats très intéressants. A la suite de ce premier circuit, un deuxième circuit de test a été conçu. Une partie est consacrée à la description de son architecture, à des simulations et à sa réalisation. Il a été implémenté dans deux technologies différentes, cependant les opportunités de fabrication que nous espérions ne se sont finalement pas concrétisées. Des simulations après le placement et le routage du circuit ont été réalisées pour explorer différentes conditions de fonctionnement du circuit en prenant en compte les parasites.

Enfin, une conclusion permet de résumer les points abordés et les résultats obtenus. Des perspectives à ce travail de thèse sont ensuite présentées.

Chapitre I.

Etat de l'art

I. 1.	Introduction.....	6
I. 2.	Les circuits asynchrones	6
I. 2. 1.	Limitations des circuits synchrones.....	6
I. 2. 2.	Conception des circuits asynchrones.....	7
I. 2. 3.	Protocoles de communication	10
I. 2. 4.	Classes des circuits asynchrones.....	13
I. 2. 5.	Avantages des circuits asynchrones.....	15
I. 3.	Rayonnement électromagnétique des circuits numériques	16
I. 3. 1.	Champ électromagnétique.....	16
I. 3. 2.	Interférences électromagnétiques.....	17
I. 3. 3.	Compatibilité électromagnétique.....	18
I. 3. 4.	Réduction du rayonnement dans les circuits synchrones.....	19
I. 3. 5.	Réduction du rayonnement dans les circuits asynchrones	22
I. 4.	Objectifs de la thèse.....	23
I. 5.	Conclusion	24

Résumé :

Les limitations de la logique synchrone, dues au signal d'horloge, rendent de plus en plus intéressante la logique asynchrone. Celle-ci permet de retirer le signal d'horloge en utilisant une synchronisation locale avec un protocole dit de poignée de main. Cette méthode de conception présente de nombreux avantages, notamment en ce qui concerne les émissions électromagnétiques du circuit.

La compatibilité électromagnétique des circuits devient une composante essentielle à prendre en compte lors de la conception. En effet, les circuits doivent fonctionner correctement dans un environnement avec de plus en plus d'objets connectés sans perturber les autres systèmes. La synchronisation locale des circuits asynchrones permet d'éviter les harmoniques du signal d'horloge dans le spectre électromagnétique du circuit et ainsi de réduire son rayonnement.

I. 1. Introduction

L'objectif de ce chapitre est de définir la logique asynchrone et son fonctionnement ainsi que le rayonnement électromagnétique des circuits et les problématiques qui y sont liées.

La première partie introduit la conception des circuits asynchrones. Tout d'abord, les limitations des circuits synchrones sont présentées. Celles-ci amènent les concepteurs à s'intéresser à de nouvelles méthodes de conception et notamment à la logique asynchrone. La conception de ces circuits est présentée ainsi qu'une porte logique spécifique nécessaire pour ces circuits, la porte de Muller. Deux protocoles de communication utilisés pour la synchronisation des données sont introduits. Nous verrons qu'il existe différentes classes de circuits asynchrones, qui ont chacune des avantages et des inconvénients. Enfin, les atouts des circuits asynchrones sont présentés.

La deuxième partie se concentre sur le rayonnement électromagnétique des circuits. Dans un premier temps la notion de champ électromagnétique est définie. Ensuite, le problème des interférences électromagnétiques, entre plusieurs systèmes, est introduit. En effet, le fonctionnement d'un circuit ne doit pas empêcher un autre circuit de fonctionner. C'est pourquoi des normes ont été établies. Nous verrons différentes méthodes pour réduire le rayonnement électromagnétique des circuits synchrones. Pour finir, des mesures électromagnétiques, sur différents circuits asynchrones, sont présentées et comparées avec les mesures des circuits synchrones équivalents.

Enfin, la dernière partie de ce chapitre, présente le contexte dans lequel s'inscrit ce travail de thèse. Les objectifs des travaux seront présentés ainsi que la démarche scientifique pour développer une solution.

I. 2. Les circuits asynchrones

I. 2. 1. Limitations des circuits synchrones

Grâce à leur simplicité de conception les circuits synchrones ont été très largement adoptés dans l'industrie de la microélectronique. Dans un système synchrone la communication entre les différentes entités est réalisée avec un signal de synchronisation global appelé signal d'horloge. Traditionnellement, dans les circuits synchrones, la logique combinatoire est placée entre des bancs de registres qui mémorisent les données et sont contrôlés par le signal d'horloge, comme montré sur la Figure I.1.

Pour assurer la validité de la donnée au moment de la mémorisation, les blocs combinatoires doivent avoir fini leurs opérations en un cycle d'horloge, peu importe les valeurs des entrées et dans le pire cas d'utilisation. Ce fonctionnement implique une contrainte temporelle forte au niveau du signal de synchronisation. En effet, la période

d'horloge doit être supérieure au temps de traitement du bloc combinatoire le plus lent. Le chemin de données, entre deux registres, le plus lent du circuit est appelé le chemin critique.

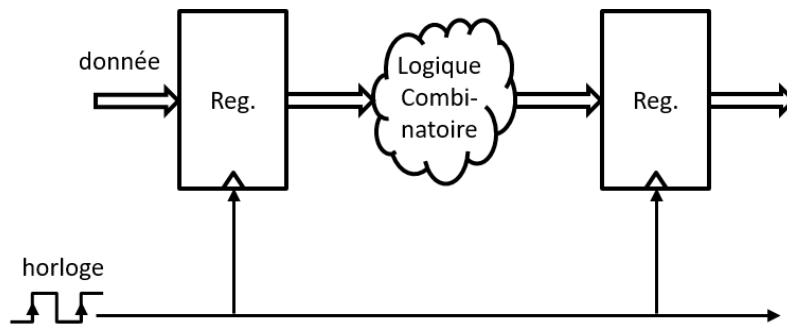


Figure I.1. Architecture d'un circuit synchrone.

De plus, le signal d'horloge doit être distribué de manière équilibrée vers tous les éléments de mémorisation afin d'éviter sur ces derniers une arrivée déphasée (du signal d'horloge). En effet, si les données ne sont pas mémorisées au même moment, des données erronées peuvent être enregistrées. Ce phénomène, de déphasage de l'horloge, est connu sous le nom de *skew*. Pour éviter ce problème, un arbre d'horloge est construit avec des buffers afin d'équilibrer tous les chemins. Ainsi, tous les éléments de mémorisation reçoivent l'horloge simultanément, d'où la notion de synchronisation globale. L'ajout de ces éléments logiques augmente la surface du circuit et ajoute une étape dans le flot de conception. De plus, la propagation de l'horloge consomme entre 20% et 45% de l'énergie totale du circuit [3], [4], et elle augmente avec la vitesse de fonctionnement du circuit. Cela rend donc plus difficile la conception des circuits basse consommation en logique synchrone.

Enfin, la périodicité du signal d'horloge crée un fort rayonnement électromagnétique au fondamental et aux harmoniques de la fréquence de fonctionnement. Nous verrons dans la partie I.3 qu'il est très difficile de diminuer ce rayonnement.

En raison des différentes limitations liées aux circuits synchrones, les concepteurs s'intéressent à une nouvelle méthode de conception basée sur la logique asynchrone. En effet, cette logique utilise une synchronisation locale des éléments de mémorisation et non globale qui permet d'éviter les contraintes liées au signal d'horloge.

I. 2. 2. Conception des circuits asynchrones

I. 2. 2. 1. Principe

Suite à l'augmentation de la complexité des circuits et du routage du signal d'horloge, de nouvelles méthodes de conception numérique sont devenues des alternatives intéressantes. La logique asynchrone est apparue dans la recherche académique aux alentours des années 1950 [5]. Cependant, il a fallu attendre les années 1990 pour que cette technologie soit mature et pour avoir les premiers circuits asynchrones [6], [7].

Dans un circuit asynchrone le séquençage des données n'est plus effectué grâce à un signal global d'horloge mais grâce à des communications locales entre les différents éléments du circuit. Ainsi, le circuit est contrôlé par des signaux internes, utilisant un protocole dit de poignée de main, qui permet d'échanger des informations entre les différents blocs, comme montré sur la Figure I.2. L'activité d'un bloc est donc liée à l'activité du bloc précédent et du suivant.

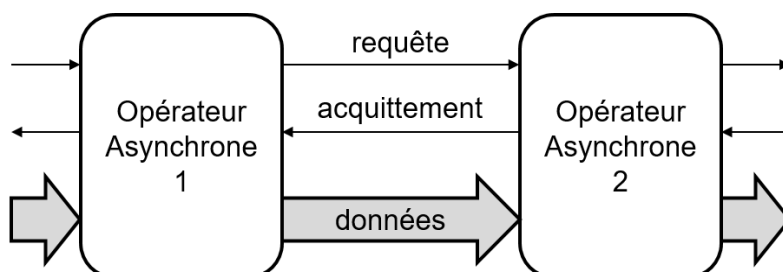


Figure I.2. Schéma fonctionnel d'un circuit asynchrone.

Lorsque l'opérateur asynchrone 1 (Figure I.2) a fini le traitement de ses données il envoie un signal de requête à l'opérateur 2, lui permettant d'indiquer que des données valides sont prêtes à ses bornes. L'opérateur 2 va alors effectuer son traitement sur les données. Ensuite, il envoie un signal d'acquiescement à l'opérateur 1 pour lui indiquer qu'il est à nouveau prêt à recevoir de nouvelles données. Il envoie également un signal de requête à l'opérateur suivant. La communication entre les opérateurs est bidirectionnelle, un échange d'informations entre les deux opérateurs est nécessaire pour l'envoi de données [8].

Le fonctionnement des circuits asynchrones est de type flot de données, chaque bloc est activé lorsqu'une donnée est présente en entrée. Cette méthode permet d'activer seulement les étages nécessaires et de laisser inactifs les autres parties du circuit, donc de consommer uniquement aux endroits où une donnée pertinente circule.

I. 2. 2. 2. Porte de Muller

En logique synchrone, les signaux sont généralement pris en compte sur les fronts montants du signal d'horloge. Ainsi, les aléas entre deux fronts montants ne sont pas gênants d'un point de vue fonctionnel, à condition qu'ils soient corrigés avant la mémorisation des données. En revanche, en asynchrone toutes les transitions des signaux peuvent être prises en compte en fonction de la classe de circuits asynchrones visée. Dans ce cas particulier, les aléas sont donc interdits [9]. Dans cette thèse, nous nous focaliserons sur des circuits dits *bundled data* dont seule la partie contrôle devra être soumise à cette règle. Le chemin de données sera, quant à lui, classique et ne présentera pas de différences avec les circuits synchrones. Ces points sont explicités plus loin dans le manuscrit.

La porte de Muller, aussi appelée C-élément, introduite par D.E. Muller [10] permet d'implémenter le réseau de contrôle du circuit. Cette porte permet de réaliser le rendez-vous entre deux signaux. Elle possède deux ports d'entrée et un port de sortie comme représenté

sur la Figure I.3.a. Elle recopie sur sa sortie la valeur de ses entrées lorsqu'elles sont identiques sinon elle mémorise la valeur précédente comme le montre la table de vérité de la Figure I.3.b.

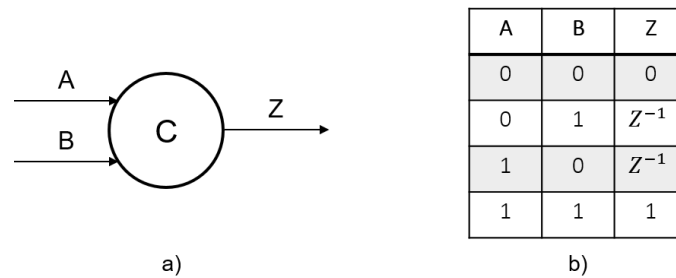


Figure I.3. a) Schéma et b) Table de vérité de la porte de Muller ou C-élément.

La Figure I.4 montre une implémentation possible de la porte de Muller avec des portes logiques standards ET et OU. Une porte « majorité » peut également être utilisée pour implémenter la porte de Muller.

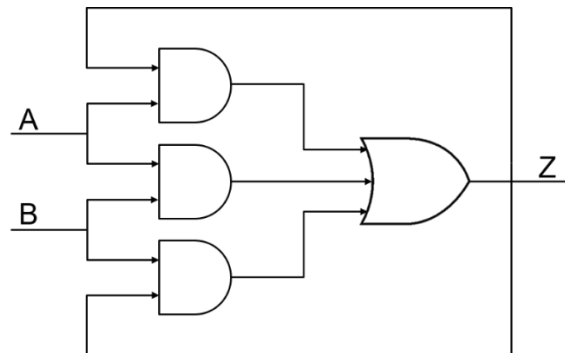


Figure I.4. Implémentation de la porte de Muller avec des portes logiques standards.

La porte de Muller permet d'effectuer le rendez-vous entre deux signaux. Elle est utilisée dans la logique asynchrone pour réaliser la synchronisation entre les signaux de contrôle, qui permettent d'échanger les données. La requête et l'acquittement arrivent sur les entrées de la porte de Muller pour pouvoir générer la requête pour l'étage suivant et l'acquittement pour l'étage précédent, comme le montre la Figure I.5. La requête d'entrée provient de l'étage précédent et permet d'indiquer que des données sont valides lorsqu'elle est au niveau logique haut. L'acquittement d'entrée est envoyé par l'étage suivant pour notifier qu'il est disponible pour recevoir des nouvelles données, lorsqu'il est au niveau logique bas. Ainsi, pour propager la donnée et acquitter l'étage précédent, il faut inverser l'acquittement en entrée, pour avoir les deux signaux au niveau logique haut, comme l'indique la table de vérité de la Figure I.3.b.

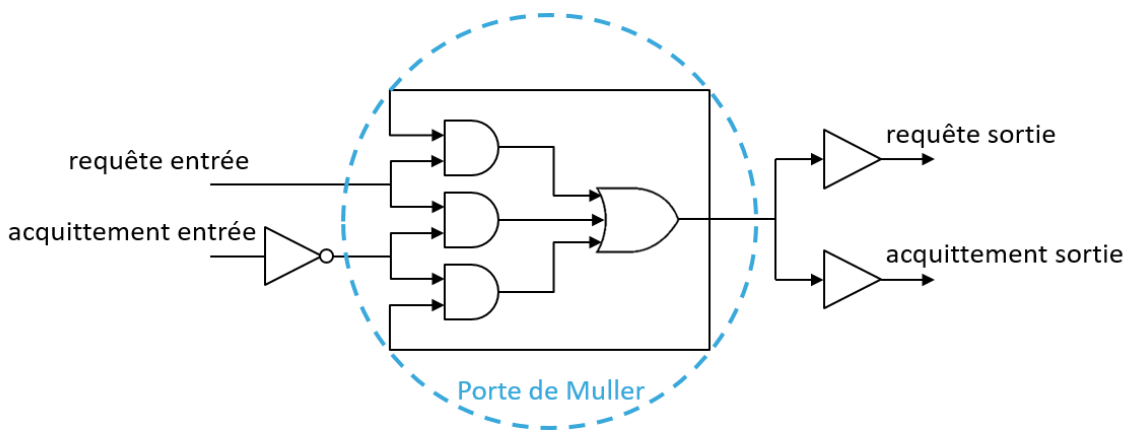


Figure I.5. Exemple d'utilisation de la porte de Muller pour les signaux de contrôle.

La porte de Muller est utilisée dans le chemin de contrôle des circuits asynchrones. Elle permet de réaliser différents protocoles de communication pour que les données circulent dans le circuit.

I. 2. 3. Protocoles de communication

Les protocoles de communication permettent la synchronisation entre les différents blocs du chemin de contrôle du circuit.

En asynchrone, les communications sont réalisées à l'aide de protocoles de type « poignée de mains », avec un signal de requête et un signal d'acquittement. Il existe des protocoles 2 phases, sensibles aux événements, et des protocoles 4 phases sensibles aux niveaux des signaux [7], [8].

I. 2. 3. 1. Protocole deux phases

Le protocole deux phases [11] aussi dit « non-retour à zéro » (NRZ) a la particularité de ne pas distinguer les transitions. Elles sont équivalentes et sont appelées événements.

Dès que l'émetteur a fini le traitement de ses données, il crée un événement sur le signal de requête. Le système entre alors en phase 1 (voir la Figure I.6). Dans cette phase, le récepteur est actif. Il traite les données et crée ensuite un événement sur le signal d'acquittement pour signifier qu'il peut recevoir de nouvelles informations. Le système bascule alors en phase 2 durant laquelle l'émetteur attend le signal d'acquittement. Une fois reçu, il peut alors renvoyer une nouvelle donnée en générant un nouvel événement sur la requête.

indique la validité de la donnée et un niveau haut du signal d'acquittement indique la consommation de la donnée. Le protocole passe obligatoirement par une phase de réinitialisation entre chaque donnée, les signaux de requête et d'acquittement repassant au niveau bas.

Lorsque l'émetteur a fini le traitement de ses données, il envoie un niveau haut sur le signal de requête au contrôleur suivant. Dans la phase 1 de la Figure I.8, le récepteur est actif. Il détecte le signal de requête et sait alors que les données sont valides. Il effectue le traitement et envoie un niveau haut sur le signal d'acquittement à l'émetteur. Dans la phase 2, l'émetteur détecte l'acquittement. Il invalide alors la donnée et passe le signal de requête au niveau bas. Ensuite, le récepteur baisse son signal d'acquittement dans la phase 3 pour signifier qu'il est prêt à recevoir de nouvelles données. Enfin, dans la phase 4, l'émetteur détecte qu'il peut à nouveau envoyer des données.

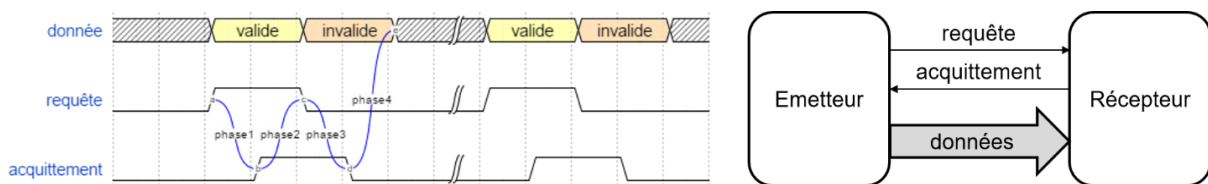


Figure I.8. Chronogramme du protocole 4-phases.

La manière la plus simple d'implémenter cette version du protocole quatre phases et d'utiliser une porte de Muller avec en entrée le signal de requête et le signal d'acquittement inversé (Figure I.9). Ainsi, pour que le contrôleur produise un 1 sur sa requête de sortie il faut qu'il ait une requête en entrée et que le contrôleur suivant soit disponible (acquittement à 0). Un délai est inséré sur le chemin de la requête pour mémoriser une donnée valide (hypothèse temporelle sur le chemin de données), comme pour l'exemple de l'implémentation du protocole 2 phases.

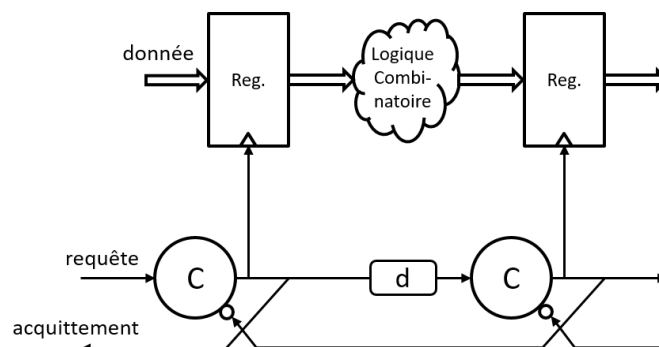


Figure I.9. Implémentation du protocole 4 phases avec une porte de Muller.

Il existe également d'autres variantes du protocole quatre phases qui ne se déclenchent pas sur les mêmes niveaux [13]. Le protocole quatre phases nécessite deux fois plus de transitions que le protocole deux phases pour transmettre une donnée. Néanmoins, des techniques d'optimisation du protocole permettent de diminuer l'impact du retour à zéro du protocole et d'avoir des circuits plus rapides en quatre phases qu'en deux phases [11].

I. 2. 4. Classes des circuits asynchrones

Il existe plusieurs classes de circuits asynchrones qui ont chacune leurs avantages et leurs inconvénients [8], [14], [15]. Elles diffèrent par rapport à la méthode d'encodage des données, du protocole de communication et du nombre d'hypothèses temporelles nécessaires au bon fonctionnement du circuit [16].

I. 2. 4. 1. Insensibles aux délais

Cette classe de circuits asynchrones est la plus robuste [17]. Il n'y a aucune hypothèse temporelle à respecter sur les délais des fils ou des portes. Un modèle de délais non bornés pour les fils et les portes logiques est utilisé. Ces circuits répondent correctement à une demande de calcul, peu importe le temps nécessaire pour l'effectuer. Il est donc nécessaire d'avoir des signaux de contrôle afin d'indiquer le début et la fin du traitement des données.

Cependant, l'utilisation du modèle de délais non bornés ne permet pas d'utiliser des portes logiques avec un seul port de sortie à l'exception de la porte de Muller [18]. En effet, une porte logique à une seule sortie peut voir sa sortie changer suite à un changement d'au moins une de ses entrées. Le processus d'acquiescement et de traitement des données n'est alors plus respecté. Il faut donc utiliser des portes complexes avec plusieurs entrées et plusieurs sorties pour réaliser un circuit de type insensible aux délais [19], ce qui en rend la conception beaucoup plus compliquée.

I. 2. 4. 2. Quasi-Insensibles aux délais

Les circuits quasi-insensibles aux délais (*Quasi-Delay Insensitive QDI*) offrent une alternative aux circuits insensibles aux délais en permettant d'autoriser les portes à une sortie [20]. Ils utilisent toujours un modèle de délais non bornés. Il n'y a pas d'hypothèses temporelles sur les fils ou les portes à l'exception des fils de type fourche isochrone. Une fourche est dite isochrone si le délai du fil entre la sortie d'une porte et les entrées des portes suivantes est identique. Cette contrainte permet de garder l'ordre des différentes transitions à l'entrée d'une porte et d'éviter les aléas.

Ces circuits utilisent un type d'encodage des données de type 1-parmi-N pour indiquer la validité des données, la requête est comprise dans la donnée. Si les N fils sont au niveau logique bas alors il n'y a pas de donnée présente. En revanche si un des fils est au niveau logique haut alors une donnée est présente et sa valeur dépend du fil qui est activé. Ainsi, les circuits quasi-insensibles aux délais sont très robustes aux variations. Néanmoins, ils nécessitent plus de fils pour encoder les données et sont généralement plus gros car la construction de ces circuits nécessite une logique sans aléas. Leur surface est donc plus importante.

I. 2. 4. 3. Indépendant de la vitesse

La conception d'un circuit indépendant de la vitesse ne fait pas d'hypothèse sur les délais des portes logiques. En revanche, les délais des fils sont considérés comme négligeables ou tous égaux [21], toutes les fourches sont ainsi isochrones. Dans les technologies récentes, les délais dans les fils étant de plus en plus longs par rapport aux portes logiques, cette contrainte devient difficile à garantir pour des grands circuits [22].

Cette classe de circuits ressemble beaucoup aux circuits quasi-insensibles aux délais, néanmoins, le fait que toutes les fourches doivent être isochrones ajoute beaucoup de vérifications. En général, les circuits quasi-insensibles aux délais sont donc préférés. Par ailleurs, les techniques de synthèse font que ces circuits sont par construction très similaires aux circuits QDI. En revanche, le modèle de circuit est lui bien différent.

I. 2. 4. 4. Micropipeline

Les circuits micropipelines ont été introduit par Ivan Sutherland en 1989 [12]. Leur fonctionnement est très proche de celui des circuits synchrones mais sans signal d'horloge global. Cette ressemblance les rend plus facile à concevoir pour des concepteurs avec une expérience en logique synchrone. Le signal d'horloge est remplacé par des contrôleurs locaux qui gèrent les signaux de requête et d'acquiescement du protocole de type poignée de main, comme représenté sur la Figure I.10. La communication entre les contrôleurs est effectuée grâce à un protocole deux phases ou quatre phases présentés précédemment dans la partie I.2.3.

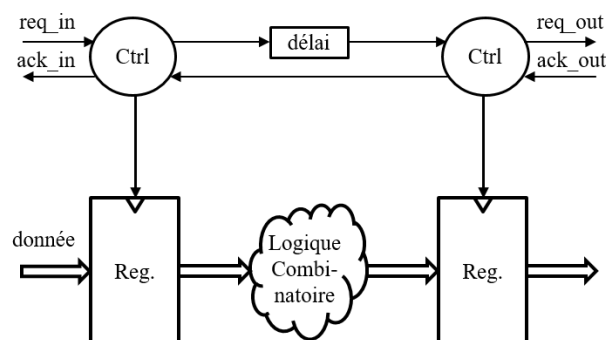


Figure I.10. Architecture d'un circuit micropipeline.

Pour assurer la validité de la donnée au moment de la mémorisation, des hypothèses temporelles sont nécessaires. En effet, le contrôleur asynchrone envoie un signal de requête au contrôleur suivant pour lui indiquer la présence de données valides à l'entrée de ses registres. Or, le signal de requête doit arriver au contrôleur suivant après la fin du traitement des données par le chemin combinatoire correspondant. Pour cela, un délai au minimum égal au délai du chemin le plus long de la logique combinatoire associée est ajouté sur le chemin de la requête, comme montré sur la Figure I.10. Le délai à ajouter est calculé entre chaque contrôleur, ce qui permet d'avoir le maximum de performance sur chaque chemin. Nous

éliminons ainsi une problématique de la logique synchrone dans laquelle la logique la plus lente ralentit tout le circuit.

I. 2. 5. Avantages des circuits asynchrones

I. 2. 5. 1. Calcul en temps minimum

En logique synchrone, le temps de calcul minimum d'un circuit est un multiple de la période de l'horloge, qui est définie par le chemin combinatoire le plus long du circuit. En revanche, avec la logique asynchrone, il n'y a plus de contrainte globale de temps. Lorsqu'un sous-bloc a fini d'effectuer son traitement sur la donnée, il en informe immédiatement le sous-bloc suivant avec le signal de requête. Ainsi, le temps de calcul du circuit dépend de la somme de toutes les opérations à effectuer. De plus, il peut être optimisé suivant la classe de circuits choisie, le protocole de communication utilisé ou le type d'encodage des données.

I. 2. 5. 2. Modularité

La synchronisation locale du circuit ainsi que les communications via deux signaux, requête et acquittement, communs à tous les opérateurs, permettent d'augmenter la modularité des circuits asynchrones. En effet, le temps de traitement d'un bloc ne dépend pas du temps de traitement des blocs précédents ou des suivants. Ainsi, la construction de fonctions complexes est plus facilement réalisable en connectant des blocs de bases entre eux. Il est très commun, dans l'industrie, de réutiliser des blocs existants plutôt que de tout redévelopper.

I. 2. 5. 3. Basse consommation

L'absence du signal d'horloge dans les circuits asynchrones permet d'enlever la consommation due à l'arbre d'horloge. De plus, la synchronisation locale du traitement des données permet de laisser inactive toutes les parties du circuit qui n'ont pas besoin de calculer, seuls les blocs qui reçoivent le signal de requête sont actifs et consomment.

De plus, la technique de réduction de la tension d'alimentation du circuit, connue pour réduire la consommation au détriment de la performance, est très facile à mettre en place pour des circuits asynchrones. La diminution de la tension d'alimentation implique une diminution du temps de propagation des signaux dans les portes logiques. Ainsi, lorsque cette méthode est implémentée dans un circuit synchrone, il est nécessaire d'ajuster la fréquence de fonctionnement du circuit pour ne pas avoir de violation de temps et mémoriser des données valides. Il faut alors refaire une analyse temporelle du circuit. Cette méthode augmente donc la complexité de conception du circuit. En revanche, dans les circuits asynchrones, le fonctionnement des blocs n'est pas lié à une contrainte temporelle, ainsi la diminution du temps de traitement des données ne va pas impacter le bon fonctionnement

du circuit. Celui-ci fonctionnera plus lentement mais sans erreurs de calculs ni violations d'hypothèses temporelles.

I. 2. 5. 4. Faible émission électromagnétique

La diminution de la consommation de courant implique une diminution du rayonnement électromagnétique. De plus, la suppression du signal périodique d'horloge permet d'éviter ses harmoniques dans le spectre. En effet, les signaux de requêtes et d'acquittements, qui peuvent être vus comme des horloges locales, peuvent arriver à tout moment en fonction de la disponibilité d'une donnée ainsi que de la latence des blocs précédents et suivants. Cet aspect aléatoire au cours du temps permet d'obtenir une consommation de courant sans les pics périodiques des circuits synchrones et de réduire les émissions.

La réduction du rayonnement électromagnétique est devenue un enjeu de plus en plus important dans la conception de circuits électroniques. Cependant, il n'existe pas beaucoup de méthodes pour réduire les émissions des circuits dès l'étape de conception. L'avantage des circuits asynchrones pour la réduction du rayonnement électromagnétique est étudié dans la partie suivante. L'utilisation des circuits asynchrones pour le contrôle, et plus seulement la réduction, du spectre électromagnétique est le cœur de ce travail de thèse.

I. 3. Rayonnement électromagnétique des circuits numériques

I. 3. 1. Champ électromagnétique

Le champ électromagnétique est le résultat d'un champ magnétique et d'un champ électrique [23]. Même s'il est imperceptible, ce champ peut avoir un impact sur la sécurité et la santé de l'Homme [22], [23]. Toute installation électrique crée un champ électromagnétique.

En effet, lorsqu'un courant i parcourt un fil conducteur, il crée un champ magnétique noté \vec{H} exprimé en Ampère par mètre (A/m). Ce champ introduit une tension sur les boucles qui sont perpendiculaires à ses lignes.

De plus, si une tension u est appliquée sur un conducteur, elle crée un champ électrique noté \vec{E} exprimé en Volt par mètre (V/m). Ce champ engendre un courant sur les fils conducteurs qui sont parallèles à ses lignes.

Lors du fonctionnement d'un circuit la tension et le courant évoluent au court du temps ce qui crée un champ électrique et magnétique. Ils forment un champ électromagnétique exprimé en Watt par mètre (W/m) ou Tesla, comme le montre la Figure I.11. Ainsi, les effets des deux champs sont combinés. Un champ électromagnétique est émis par le conducteur qui se comporte comme une antenne. Celui-ci, n'étant pas voulu ni contrôlé, peut poser des problèmes sur l'environnement du circuit.

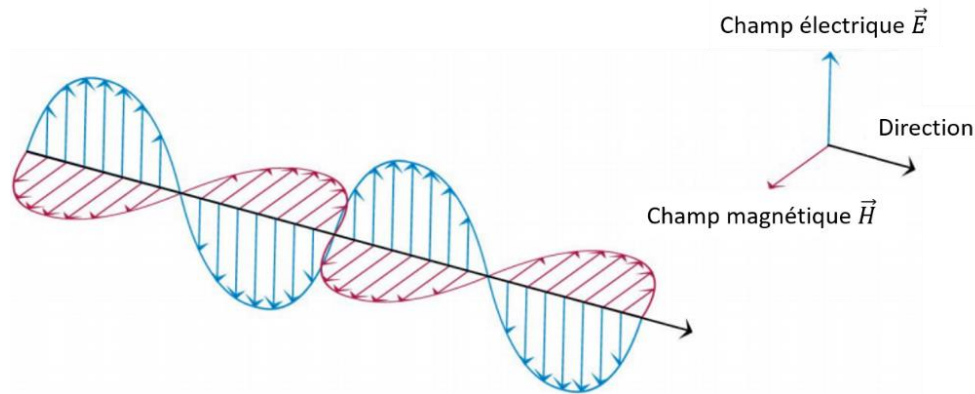


Figure I.11. Représentation du champ électromagnétique.

I. 3. 2. Interférences électromagnétiques

Le champ électromagnétique, créé lors du fonctionnement du circuit, peut interférer avec d'autres champs présents dans son environnement [26]. Les conséquences de ces interférences peuvent être graves. En effet, il peut venir perturber un champ utilisé pour une communication sans fils. Il peut également perturber ou détruire d'autres systèmes de son environnement et également perturber son propre fonctionnement.

Nous parlons d'interférences électromagnétiques lorsqu'une source émet un champ électromagnétique qui vient perturber le fonctionnement ou vient détruire un système de son environnement, qui est alors qualifié de « victime » (Figure I.12). Nous parlons d'interférences intra-système si la source et la victime font partie du même système qui s'auto-perturbe. Lorsque la source et la victime sont dans des systèmes distincts nous parlons d'interférences inter-systèmes [26].

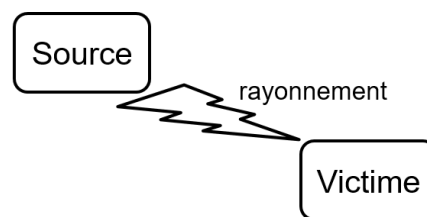


Figure I.12. Phénomène d'interférence électromagnétique.

Lorsqu'un circuit est en fonctionnement, il émet un rayonnement électromagnétique non intentionnel. Il devient alors une source de bruit qui vient polluer son environnement. Ce même circuit peut voir son fonctionnement perturbé à cause de son environnement électromagnétique, il est alors une victime. Un circuit est donc caractérisé par son émission électromagnétique et sa susceptibilité aux ondes présentes.

Les émissions parasites des circuits augmentent avec la complexité des circuits, une plus grande intégration dans les circuits ainsi que l'accélération des vitesses de commutation. L'immunité aux interférences des circuits diminue avec la réduction des tensions

d'alimentation et l'augmentation du nombre d'interfaces [27]. Pour éviter ou minimiser ces interférences des normes ont été mises en place.

I. 3. 3. Compatibilité électromagnétique

Les études sur la compatibilité électromagnétique (CEM) ont commencé dans le milieu militaire dans les années 1965, pour étudier l'effet du champ électromagnétique d'une explosion nucléaire sur les circuits électroniques dans les sites de lancement des missiles [27]. La compatibilité électromagnétique des circuits est un enjeu très important du fait de l'augmentation des objets connectés.

Un système est dit compatible avec son environnement lorsqu'il fonctionne correctement et qu'il n'empêche pas les autres systèmes présents de fonctionner correctement. Pour éviter les problèmes d'interférences la Commission Electrotechnique Internationale a défini des normes pour encadrer les émissions des systèmes ainsi que l'immunité aux rayonnements [28]. Pour cela, le Comité International Spécial des Perturbations Radioélectriques (CISPR) a été créé en 1934 pour définir des normes limitant les perturbations électromagnétiques. Ils définissent également des méthodes pour mesurer le rayonnement et la susceptibilité des circuits. En 1934, le congrès américain a créé la Commission Fédérale des Communications (*Federal Communications Commission FCC*), une agence indépendante du gouvernement des Etats-Unis, chargée de réguler les télécommunications. Cette commission a par la suite publié des normes sur les émissions électromagnétiques [29].

Depuis le 1^{er} janvier 1996, tous les circuits doivent respecter des normes de compatibilité électromagnétique afin de pouvoir être commercialisés [28], [29]. Actuellement, la CEM est devenue une des principales raisons de changement d'architecture des circuits. De plus, les problèmes de compatibilité des circuits sont généralement constatés après une première fabrication ou un prototypage du circuit, ce qui engendre des coûts très importants. Cela est principalement dû à un manque de méthodes d'analyses et d'expertises dans la réduction et le contrôle des bruits parasites [27].

Lorsqu'un circuit à un problème de compatibilité électromagnétique, il y a trois méthodes générales pour réduire les problèmes posés par les émissions. Premièrement, il est possible de supprimer l'émission à la source ou d'essayer de diminuer les émissions. Pour cela, des filtres sur la carte d'application peuvent être ajoutés. Il est également possible d'augmenter les temps de montée et de descente des signaux. En effet, les variations rapides des signaux sont une des principales causes des émissions. Plus ces temps sont rapides, plus le spectre est étendu. Cette méthode est efficace lorsqu'il n'y a pas beaucoup de signaux dans le circuit.

La deuxième solution pour réduire les problèmes des interférences est de diminuer l'efficacité du canal de transmission. Le bon sens voudrait qu'on se concentre sur la suppression de la source des émissions, car une source peut nuire à beaucoup de victimes. Cependant, cela n'est pas possible pour des sources naturelles de notre environnement,

comme par exemple la foudre. Ainsi, il est également nécessaire d'augmenter l'immunité des circuits ou d'agir sur le chemin de couplage [26]. Pour cela, le système peut être placé dans une boîte métallique pour augmenter le blindage.

Enfin, la troisième possibilité pour réduire l'impact des émissions est de rendre moins susceptible le récepteur, en isolant le circuit sur un plan de masse pour ramener les interférences à la masse ou alors en ajoutant des filtres.

I. 3. 4. Réduction du rayonnement dans les circuits synchrones

Dans un circuit synchrone, les impulsions du signal d'horloge sont responsables jusqu'à 70% de la dissipation de puissance dynamique à cause des fréquences élevées et du nombre de registres activés [30]. Ces impulsions rendent très difficile la réduction du rayonnement électromagnétique des circuits synchrones.

Normalement, la transformée de Fourier d'un signal périodique carré est dite impaire, c'est-à-dire, ses harmoniques paires sont nulles. Cela est dû à la demi période nulle du signal (Figure I.13.a). Cependant, dans un circuit numérique, de l'énergie est dissipée durant les fronts montants et descendants du signal d'horloge. Ainsi toutes les harmoniques rayonnent et nous retrouvons aussi bien les harmoniques paires que les impaires [31] comme le montre la Figure I.13.b.

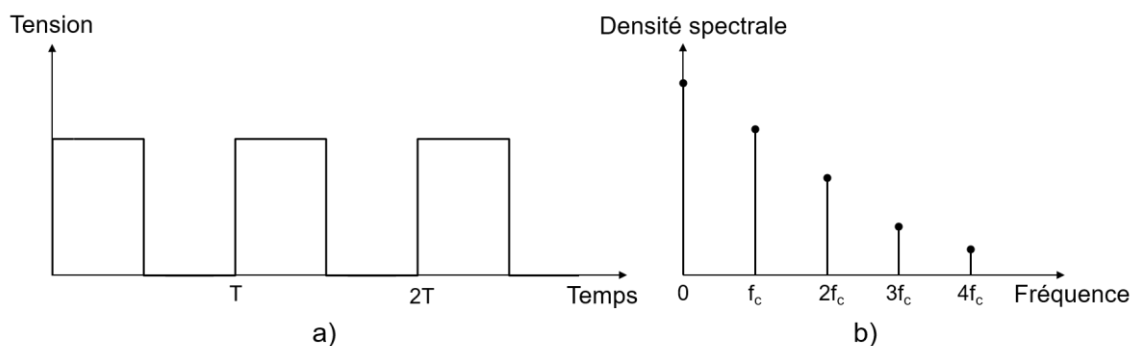


Figure I.13. Allure a) du signal d'horloge dans le temps où T est la période du signal et b) de sa densité spectrale où f_c est la fréquence du signal ($f_c = 1/T$).

Aujourd'hui, il existe quelques méthodes pour essayer de réduire ces harmoniques [32]. Ces méthodes, décrites ci-après, jouent essentiellement sur le contrôle du signal d'horloge.

I. 3. 4. 1. Désactivation de l'horloge

La première méthode consiste à désactiver le signal d'horloge d'un sous-système du circuit lorsqu'il n'a pas besoin de fonctionner. Elle est connue sous le nom de « *clock gating* » [30], [33]–[35]. Un signal vient couper l'horloge lorsque la donnée produite par le sous-circuit n'est pas intéressante et vient le redémarrer lorsqu'une donnée pertinente doit être traitée. Cette technique peut être implémentée de plusieurs manières possibles [33]. La plus simple utilise

une porte logique, telle qu'une porte ET ou NOR. Dans ce cas-là, l'implémentation est petite mais sensible aux aléas. Il est aussi possible de réaliser une cellule de désactivation de l'horloge plus robuste mais plus grosse avec un verrou (*latch* en anglais) et une porte ET, comme sur la Figure I.14. La partie qui sert à contrôler le signal d'horloge est entourée en pointillés sur la Figure I.14. Le verrou permet de mettre à jour le signal d'activation à chaque fois que l'horloge est au niveau logique bas. Ensuite un ET logique est réalisé entre le signal d'activation et l'horloge pour créer la nouvelle horloge qui permet de mémoriser les données du circuit. L'inconvénient de l'utilisation d'un verrou est sa période de pause, lorsque l'horloge est au niveau logique haut, durant laquelle le signal en sortie n'est pas mis à jour. En effet, nous risquons de louper un changement du signal d'activation [36].

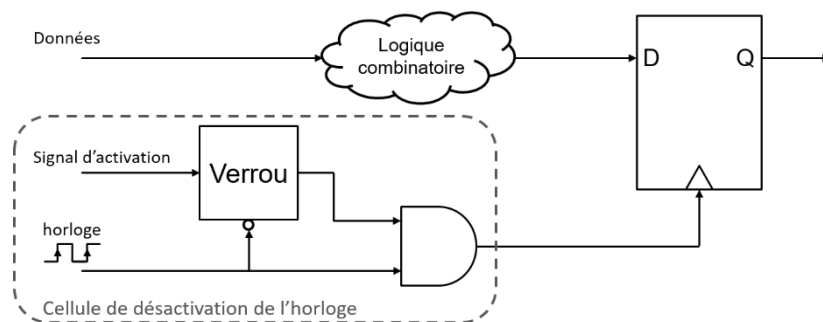


Figure I.14. Exemple de cellule de désactivation du signal d'horloge [34].

La technique de désactivation de l'horloge permet de supprimer la consommation de l'arbre d'horloge, des registres du sous-système ainsi que de la logique combinatoire associée lorsque le signal est coupé. Cependant, cette technique de conception ajoute des portes au circuit pour pouvoir arrêter et redémarrer le signal d'horloge. Une étape importante consiste à trouver le meilleur endroit dans le circuit où implémenter la cellule de désactivation de l'horloge afin de réduire au mieux la consommation. Il faut également créer le signal d'activation de l'horloge.

Une unité arithmétique et logique sur 8-bits avec 16 opérations possibles a été implémentée avec et sans méthode de désactivation d'horloge sur un circuit logique programmable (*Field-Programmable Gate Array FPGA*) [37]. Ils ont mesuré le pourcentage de consommation de l'horloge pour différentes fréquences d'opérations avec et sans l'implémentation de la méthode. A 10 GHz, la consommation de puissance de l'horloge est de 51,30% de la consommation dynamique totale. Avec la méthode de désactivation de l'horloge, la consommation de l'horloge est réduite à 23,39% de la consommation dynamique totale. Ils ont également mesuré une réduction de 37,84% du courant dynamique en utilisant cette technique [37].

Cette méthode de désactivation de l'horloge permet bien de réduire le rayonnement électromagnétique des circuits cependant, il n'est pas possible de quantifier la réduction avant de pouvoir la mesurer. Cette méthode ne permet donc pas d'assurer que le circuit va respecter les normes de CEM.

I. 3. 4. 2. Génération d'horloge à spectre étalé

La méthode d'étalement du spectre d'horloge pour les circuits intégrés a été introduite en 1994 par K.B. Hardin [38]. Cette méthode permet de réduire les émissions d'un système électronique en appliquant une modulation en fréquences sur le signal d'horloge. Sans modulation, une harmonique du signal d'horloge est représentée par une fonction delta centrée à une fréquence n fois la fréquence fondamentale du signal d'horloge. Avec la modulation, l'énergie à chaque harmonique est étalée sur une plus grande plage de fréquences et avec une amplitude réduite (Figure I.15.a). De plus, la plage de fréquences sur laquelle chaque harmonique est étalée augmente linéairement avec la fréquence.

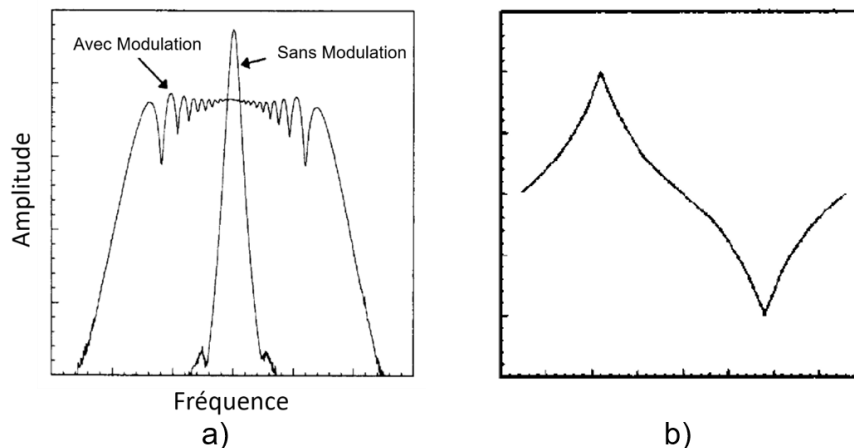


Figure I.15. a) Représentation d'un harmonique du signal d'horloge avec et sans générateur d'horloge à spectre étalé b) Représentation de la fonction de modulation [38].

Différentes fonctions peuvent être utilisées pour effectuer la modulation du signal d'horloge. Il a été montré que la fonction représentée sur la Figure I.15.b permet d'obtenir une atténuation optimale [38], uniforme sur la plage de fréquences (Figure I.15.a). La modulation peut être appliquée par un circuit externe mais elle peut aussi être intégrée à l'intérieur du circuit.

L'atténuation mesurée grâce à cette méthode est d'approximativement 2 dB au troisième harmonique et augmente jusqu'à environ 10 dB à 400 MHz pour une horloge de 20 MHz [38]. L'atténuation du $n^{\text{ième}}$ harmonique d'horloge est définie par le rapport entre l'amplitude du $n^{\text{ième}}$ harmonique du signal d'horloge non modulé et l'amplitude du $n^{\text{ième}}$ harmonique du signal d'horloge modulé. Cette méthode a également été appliquée pour réduire les émissions d'un convertisseur de tension pour éviter les interférences électromagnétiques [39]. Ils ont montré une réduction de 5,7 dBm pour l'harmonique principale, 15,6 dBm et 12,8 dBm pour respectivement le deuxième et troisième harmonique.

Cette méthode permet de réduire les harmoniques dues au signal d'horloge, cependant, elle ne permet pas de les supprimer. De plus, il faut rajouter une modulation en fréquence de l'horloge dans le circuit.

I. 3. 5. Réduction du rayonnement dans les circuits asynchrones

Par construction, les circuits asynchrones ont un spectre électromagnétique plus étalé. En effet, l'absence d'horloge permet de ne pas avoir les harmoniques de ce signal dans le spectre. Les appels de courant des commutations des registres ne sont plus simultanés. L'avantage des circuits asynchrones a été prouvé avec des mesures sur différents circuits [40]–[45]. Cette partie présente trois exemples.

Philips Semi-conducteurs a montré que la version asynchrone du microcontrôleur 80C51 permet de réduire le spectre électromagnétique par rapport à la version synchrone [41]. Le spectre électromagnétique du 80C51 asynchrone ne présente plus les harmoniques à la fréquence d'horloge contrairement aux mesures du spectre du microcontrôleur synchrone [46]. Pour certaines fréquences le spectre est réduit d'environ 30 dB dans la version asynchrone (Figure I.16).

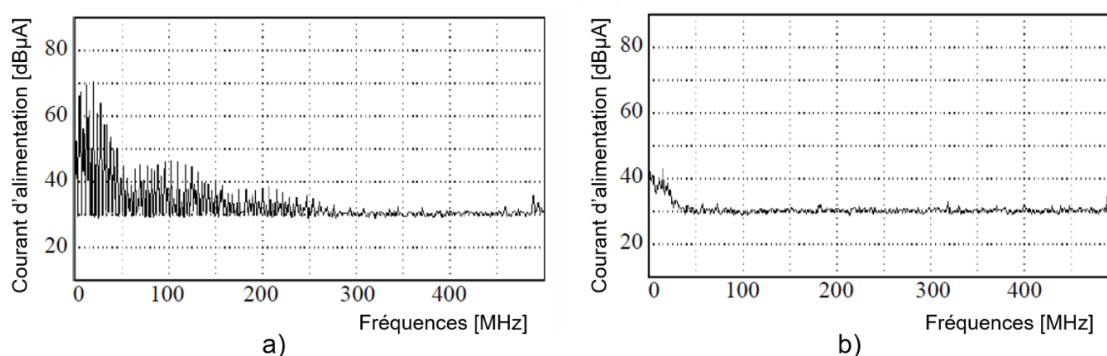


Figure I.16. Spectre électromagnétique du courant d'alimentation mesuré sur la version a) synchrone et b) asynchrone du microcontrôleur 80C51 [46].

De plus, ils ont montré que la version asynchrone consomme environ quatre fois moins de puissance que la version synchrone avec le même environnement de fonctionnement. Ces propriétés ont permis à Philips Semi-conducteurs d'intégrer le microcontrôleur 80C51 asynchrone dans des cartes à puce dont le microcontrôleur pouvait rester actif pendant la réception des messages. En effet, il n'y avait plus d'harmoniques qui interféraient pendant la communication. Cela n'était pas possible avec le microcontrôleur synchrone [46].

A l'université de Manchester, un système embarqué, AMULET2e, contenant un processeur asynchrone compatible ARM 32-bit a été développé [42]. Ils ont montré que son spectre électromagnétique est plus faible que celui de son équivalent synchrone, le microcontrôleur ARM60 PIE. Le rayonnement de l'AMULET2e présente encore des harmoniques dues à la périodicité du logiciel du système. Néanmoins, les harmoniques sont plus faibles que dans la version synchrone comme le montre la Figure I.17.

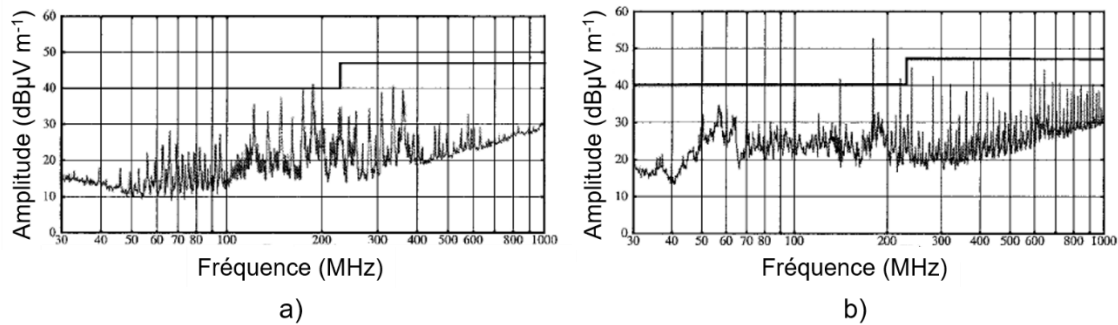


Figure I.17. Spectre électromagnétique du système embarqué a) asynchrone AMULET2e et b) synchrone ARM60 PIE [42].

Enfin, les performances électromagnétiques de la logique quasi-insensible aux délais (QDI) ont également été évaluées. En effet, un crypto-processeur a été implémenté en asynchrone et comparé avec sa version synchrone [43]. La version QDI rayonne 5,6 fois moins dans la bande de fréquences 0-200 MHz où l'activité principale se trouve, comme le montre la Figure I.18. De plus, la version asynchrone fonctionne 27 fois plus rapidement que la version synchrone.

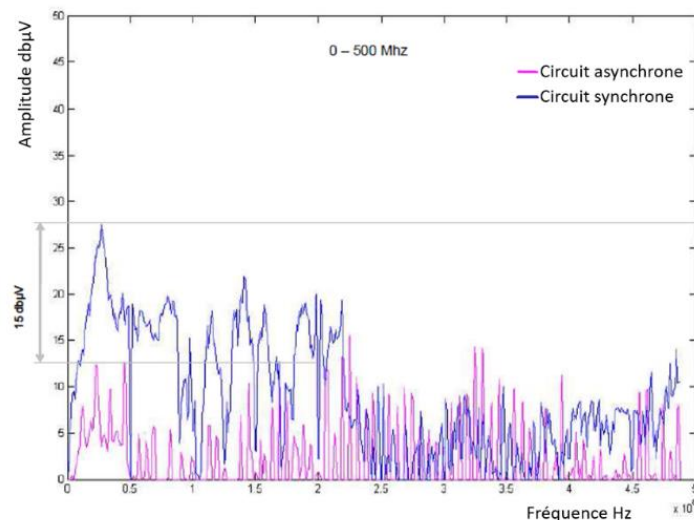


Figure I.18. Comparaison du spectre électromagnétique du crypto-processeur synchrone en bleu et asynchrone en rose à 1,8 V sur la plage de fréquences 0-500 MHz.

Ces différentes mesures montrent les propriétés intéressantes et les avantages des circuits asynchrones pour répondre aux normes de compatibilité électromagnétique, sans effort de conception pour limiter le rayonnement des circuits.

I. 4. Objectifs de la thèse

Les travaux de cette thèse s'intéressent à la conception des circuits intégrés dans le cadre de la compatibilité électromagnétique (CEM). En effet, nous avons vu dans ce premier chapitre que la CEM est un enjeu majeur pour la conception, car elle est devenue une des premières

causes de changement d'architecture des circuits [27]. Or, cela coûte du temps et de l'argent de refaire un circuit. Il existe des méthodes qui permettent de réduire ou d'étaler le rayonnement des circuits synchrones. Une autre solution consiste à utiliser des circuits asynchrones qui, par construction, permettent de réduire les émissions. Toutes ces méthodes ont pour limite de réduire de façon qualitative les émissions mais ne permettent pas de les contrôler.

Néanmoins, réduire le spectre électromagnétique n'assure pas le respect des normes CEM imposées pour pouvoir commercialiser un produit. C'est pourquoi, la définition d'un gabarit spectral est nécessaire dès l'étape de conception d'un circuit. De plus, le gabarit permet au concepteur de définir des bandes de fréquences à réduire ou à déplacer, en fonction de l'application. Notre méthode permet d'ajuster les émissions en fonction du gabarit demandé lors de l'étape de conception. Elle est détaillée dans le chapitre II.

Pour obtenir le spectre électromagnétique du circuit dès l'étape de conception, une transformée de Fourier de la consommation de courant est effectuée. Ainsi, pour pouvoir estimer le rayonnement du circuit, il est nécessaire d'avoir une modélisation de la consommation de courant. Pour cela, nous avons réalisé un outil de simulation analogique rapide qui nous permet de connaître les instants de commutation des registres du circuit. Il nous permet ensuite d'associer à chaque instant de commutation le pic de consommation associé. La méthode de modélisation de haut niveau du courant est présentée dans le chapitre III.

Cet outil a été intégré dans un flot de conception qui permet de contrôler le rayonnement grâce à un gabarit spectral. Ainsi, les paramètres à changer dans le circuit sont évalués automatiquement dans le flot. L'articulation des différentes étapes du flot de conception est décrite dans le chapitre IV.

Enfin, le chapitre V présente l'évaluation de notre méthode de conception avec deux circuits de test. Deux architectures différentes ont été réalisées avec notre flot de conception dont une fabriquée sur silicium en technologie CMOS 40 nm STMicroelectronics. Des tests ont été effectués pour mesurer le champ électromagnétique du circuit avec notre méthode de contrôle du rayonnement.

I. 5. Conclusion

Dans la première partie de ce chapitre, nous avons vu que la logique synchrone présente des limitations pour les nouveaux enjeux de conception, notamment à cause du signal d'horloge. En effet, celui-ci est difficile à router dans des circuits complexes et introduit des harmoniques périodiques, difficiles à réduire, dans le spectre électromagnétique. Pour pallier à ces problèmes, la logique asynchrone est très intéressante. L'utilisation de contrôleurs locaux pour faire avancer le traitement des données permet de s'affranchir des contraintes liées au signal d'horloge. Un protocole de communication est utilisé pour faire avancer les

données. Différentes classes de circuits asynchrones existent avec plus ou moins d'hypothèses temporelles et de difficultés de conception. Dans ce manuscrit, nous utiliserons la logique micropipeline. Néanmoins, notre technique peut être utilisée sur n'importe quelle autre classe de circuits asynchrones.

Dans la seconde partie, nous avons vu que le rayonnement électromagnétique des circuits est de plus en plus important dans la conception des produits électroniques. En effet, si cette étape n'est pas maîtrisée le circuit crée de la pollution électromagnétique qui perturbe le fonctionnement ou détruit les autres systèmes présents dans son environnement. Pour éviter ce genre de problèmes, des normes électromagnétiques ont été établies et doivent être respectées par tous les circuits commercialisés. Les problèmes de compatibilité électromagnétique sont devenus une des premières causes de changement d'architecture des circuits. Nous avons vu qu'il existe des méthodes pour essayer de réduire le spectre électromagnétique des circuits synchrones. Cependant, les harmoniques présentes à cause du signal d'horloge sont très difficiles à réduire. La logique asynchrone, par construction, permet de ne plus avoir les harmoniques dans le spectre. Elle a fait ses preuves pour réduire le rayonnement à travers les mesures de plusieurs circuits.

Enfin, la problématique du contrôle du rayonnement électromagnétique dès l'étape de conception a été soulevée. L'ajout du respect des normes de compatibilité électromagnétique dans les spécifications de la conception d'un produit permettrait de gagner beaucoup de temps et d'argent. La démarche scientifique pour créer un flot de conception permettant de répondre à cette problématique a été énoncée. Le principe de la méthode est décrit dans le chapitre suivant.

Chapitre II.

Principes du contrôle du rayonnement

II. 1.	Introduction.....	28
II. 2.	Premières tentatives de contrôle du spectre.....	28
II. 2. 1.	Méthode synchrone	29
II. 2. 2.	Méthode asynchrone	32
II. 3.	Introduction d'un gabarit	34
II. 3. 1.	Principe.....	34
II. 3. 2.	Deux Exemples de Normes.....	35
II. 4.	Présentation de notre méthode.....	37
II. 4. 1.	Hypothèse pour l'émission électromagnétique	37
II. 4. 2.	Ajout de délais.....	38
II. 4. 3.	Détermination des valeurs de délais.....	40
II. 5.	Conclusion	44

Résumé :

Actuellement, un circuit doit obligatoirement respecter des normes électromagnétiques pour être commercialisé. Il existe des méthodes de mise en forme du courant pour réduire le rayonnement électromagnétique des circuits numériques. Cependant, nous ne trouvons pas aujourd'hui de méthode pour contrôler les émissions et s'assurer qu'elles respectent des contraintes en amplitude.

Notre méthode prend en compte, dès l'étape de conception, un gabarit en fréquences, qui peut être issu d'une norme ou modifié par le concepteur. Le spectre du circuit est directement déduit de la consommation en courant. L'objectif est de déterminer les instants des appels de courant pour obtenir un spectre qui respecte le gabarit. Bien que toutes les classes de circuits asynchrones puissent être utilisées, nous utiliserons préférentiellement les circuits micropipelines, en ajoutant des délais supplémentaires sur le chemin du signal de requête. Ce contrôle permettra de déplacer les appels de courant. Les valeurs des délais à insérer sont déterminées grâce à un algorithme génétique.

II. 1. Introduction

L'objectif de ce chapitre est de présenter notre méthode de contrôle du rayonnement électromagnétique des circuits.

Pour cela, la première partie présente les premières tentatives de contrôle du rayonnement électromagnétique dans les domaines synchrones et asynchrones. En effet, dans le chapitre précédent nous avons vu des méthodes synchrones pour réduire le rayonnement ainsi que l'avantage que pouvait apporter des circuits asynchrones sans effort de conception particulier. Nous allons maintenant détailler une méthode qui permet de réduire le rayonnement sur une plage de fréquences précise pour un circuit numérique synchrone. Nous verrons également une méthode de conception asynchrone, ayant pour but la réduction des émissions électromagnétiques.

L'objectif de la deuxième partie est d'introduire l'utilisation d'un gabarit fréquentiel pour contraindre le rayonnement des circuits. Il existe de nombreuses normes spécifiques pour les applications des produits qui décrivent les limites acceptées pour les émissions électromagnétiques. Il est donc intéressant d'utiliser ces gabarits pour contraindre la conception des circuits et assurer leur bon fonctionnement. De plus, ce gabarit peut être ajusté par le concepteur pour répondre à des besoins spécifiques du circuit.

Enfin, la troisième partie de ce chapitre présente notre démarche pour contrôler le rayonnement électromagnétique des circuits à partir d'un gabarit, présenté dans la deuxième partie. Nous faisons l'hypothèse que les émissions électromagnétiques sont directement liées aux appels de courant des circuits. C'est pourquoi, l'objectif est de déplacer les instants des appels de courant dus aux éléments de mémorisation de manière à avoir un spectre qui respecte le gabarit en tout point. Pour décaler temporellement les appels de courant, des délais sont introduits sur le chemin du signal de requête entre les contrôleurs du circuit. Pour finir, notre méthode pour déterminer les valeurs des délais est présentée. Une approche algorithmique nous permet de trouver une combinaison de délais afin que le spectre du circuit soit en dessous du gabarit.

II. 2. Premières tentatives de contrôle du spectre

Les méthodes et circuits synchrones présentés dans le Chapitre I permettent de réduire ou d'étaler le spectre mais pas de le contrôler. Les mesures des circuits asynchrones présentées dans le Chapitre I confirment que cette méthode de conception permet de réduire et d'étaler le spectre.

Cette partie présente deux méthodes : une en synchrone, qui a pour but de diminuer le spectre électromagnétique de certaines plages de fréquences, et une en asynchrone, qui permet de réduire le spectre. Ces deux méthodes ont en commun de mettre en forme le courant pour réduire le bruit créé par les circuits numériques.

II. 2. 1. Méthode synchrone

L'objectif de cette méthode [47] est de planifier le signal d'horloge de manière à optimiser le bruit engendré par les circuits numériques sur des bandes de fréquences ciblées.

Il a été montré dans [48] que le spectre de la consommation de courant peut être décomposé en deux parties : des pics discrets et un bruit de fond continu. Les pics correspondent au signal périodique d'horloge qui cadence les traitements du circuit alors que le bruit de fond continu correspond aux fluctuations entre les fronts d'horloge. Or, les pics sont largement au-dessus du bruit de fond et la plupart de la puissance est concentrée dans ces pics. C'est pourquoi, dans cette méthode d'optimisation seuls les pics de courant dus au signal d'horloge sont considérés [47]. Pour modéliser la consommation de courant, une forme triangulaire est utilisée pour les pics [48], comme montré sur la Figure II.1.

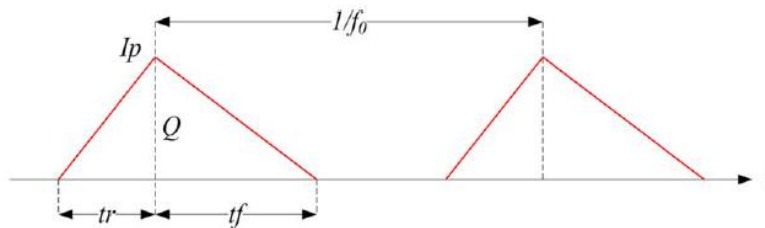


Figure II.1. Modèle triangulaire de la consommation de courant dans un circuit synchrone avec I_p l'amplitude du pic, t_r le temps de montée, t_f le temps de descente, f_0 la fréquence de l'horloge et Q l'aire du triangle [47].

Dans cette méthode, la réduction du spectre, dans les hautes fréquences, est obtenue en étalant la consommation de courant dans le temps. C'est-à-dire, en diminuant I_p et en augmentant t_r et t_f pour garder la même aire pour le triangle (voir la Figure II.1). Ils ont montré en simulations numériques qu'en augmentant cinq fois la largeur du pic de courant une diminution de 28 dB du spectre dans les hautes fréquences est obtenue (Figure II.2). Cependant, la diminution du spectre est de seulement 6 dB à la fréquence d'horloge. Ainsi, la méthode d'étalement du courant est efficace pour les hautes fréquences mais pas pour les basses.

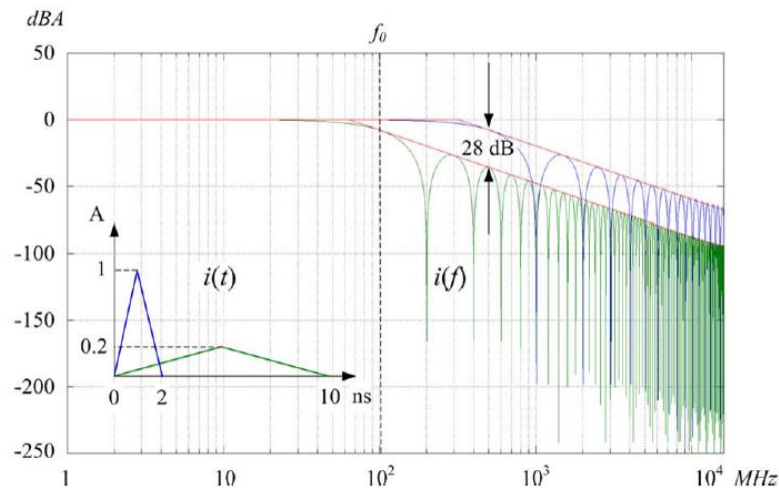


Figure II.2. Simulations numériques de la méthode d'étalement du courant dans un circuit synchrone. Deux profils de courant sont étudiés pour un circuit fonctionnant à 100 MHz. Un triangle à peine étalé et son spectre en bleus, sont comparés à un triangle 5 fois plus étalé et son spectre en verts. La ligne rouge représente l'enveloppe des deux spectres [47].

Pour obtenir une réduction des émissions dans les basses fréquences, ils proposent d'utiliser les deux fronts du signal d'horloge [47]. Le circuit est divisé en deux domaines d'horloges différents, un sur les fronts montants et l'autre sur les fronts descendants. De plus, l'étalement des pics de courant peut être optimisé pour chaque domaine d'horloge. Ainsi, il est aussi possible de réduire les hautes fréquences du spectre en utilisant les deux fronts de l'horloge. Pour garder la même latence sur les données que dans un circuit synchrone classique, Figure II.3.a, un verrou est inséré avant le dernier étage pour ajouter une demi-période d'horloge à la latence, le verrou E sur la Figure II.3.b. Les bascules B et C mémorisent sur les fronts descendants de l'horloge tandis que les bascules A et D sont sensibles aux fronts montants. La prise en compte des deux fronts de l'horloge complexifie l'analyse temporelle statique du circuit et les étapes de vérification.

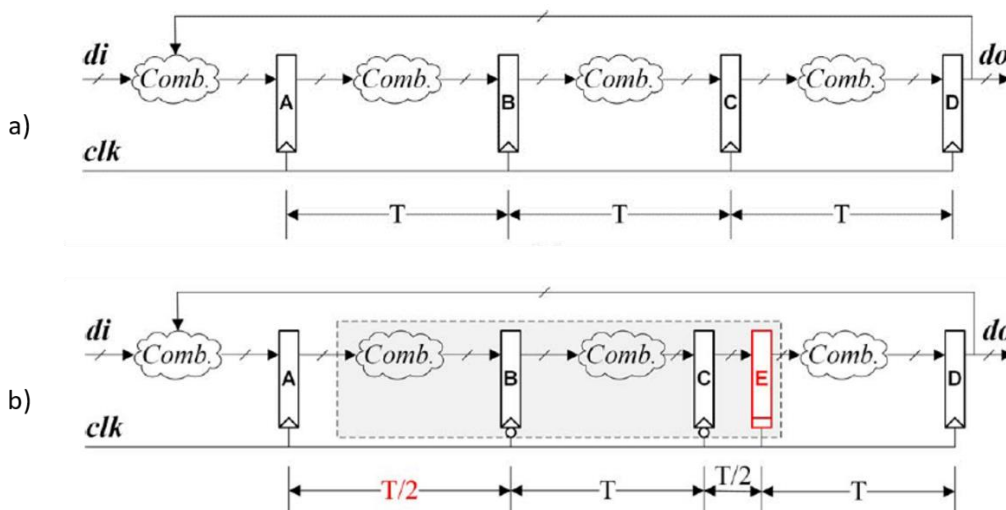


Figure II.3. a) Représentation d'un pipeline synchrone utilisant les fronts montants de l'horloge et b) d'un pipeline utilisant les deux phases de l'horloge avec l'ajout d'un verrou de données avant le dernier étage [47].

Enfin, pour optimiser le spectre sur une bande de fréquences spécifiée par l'utilisateur, ils ont développé un flot de conception, dans lequel ils ajoutent une optimisation de la latence d'horloge [47]. Décaler l'instant d'arrivée du signal d'horloge sur les bascules permet de réduire le spectre sur des plages de fréquences spécifiques. Un système d'optimisation est mis en place dans lequel la réponse en fréquence et l'estimation de la taille de l'arbre d'horloge sont pris en compte. Cette nouvelle étape est insérée dans le flot après le placement du circuit et avant la construction de l'arbre d'horloge. Le flot a besoin des informations extraites du placement pour concevoir l'arbre d'horloge de façon efficace.

Cette méthode a été évaluée sur un Nœud de Capteur de Confiance (*Trusted Sensor Node TSN* en anglais) [47]. La fréquence principale est de 50 MHz et une plage de fréquences autour des 850 MHz est visée pour l'optimisation. Cette plage est utilisée pour les communications sans fils des capteurs. Ce circuit est implémenté en utilisant quatre méthodes différentes, chacune appelée macro. La macro 0 est sans optimisation, les macros 1 et 2 utilisent la méthode d'étalement du courant avec une optimisation plus agressive pour le 2 et enfin la macro 3 utilise les deux fronts du signal d'horloge. La Figure II.4 montre les résultats des mesures des interférences électromagnétiques des 4 macros. Nous constatons que le circuit sans optimisation (macro 0) a une émission beaucoup plus forte dans la plage de fréquences autour de 850 MHz. La macro 2, avec son optimisation agressive de la latence du signal d'horloge, a le meilleur résultat sur cette plage de fréquences. En revanche, pour la fréquence principale de 50 MHz, la macro 3 a la meilleure performance, grâce à l'utilisation des deux fronts de l'horloge.

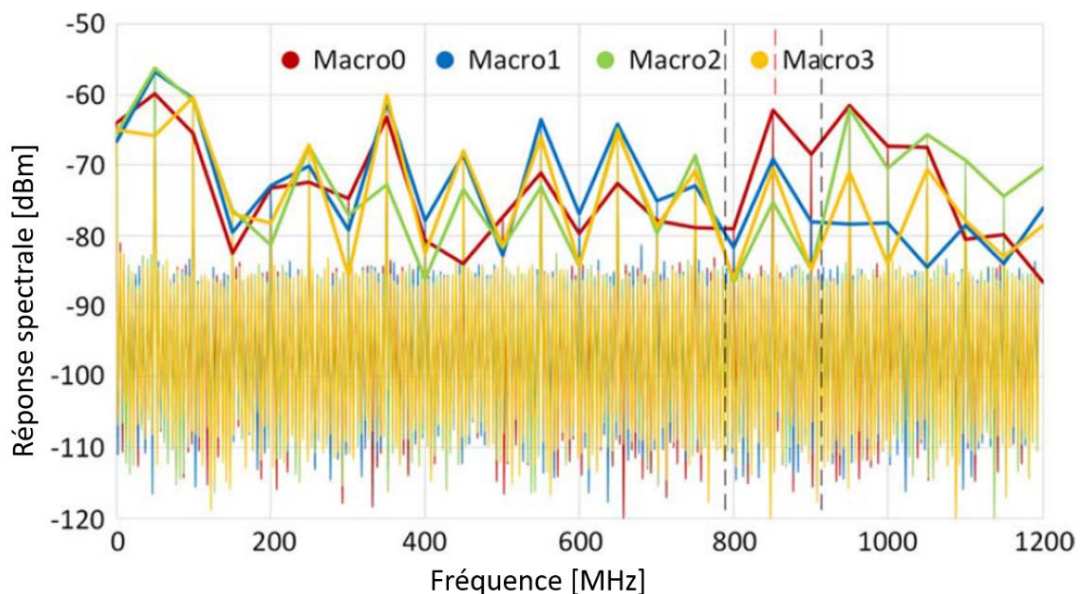


Figure II.4. Superposition des interférences électromagnétiques conduites des 4 macros [47].

Les résultats de cette méthode montrent qu'elle permet de réduire le spectre électromagnétique des circuits numériques en étalant la consommation de courant. Elle permet également d'optimiser une plage de fréquences spécifique. Néanmoins, l'étalement de la largeur des pics en courant est limité par les contraintes temporelles et limite donc les

performances de cette méthode [49]. De plus, l'utilisateur ne peut pas définir l'amplitude maximale autorisée pour les plages de fréquences ciblées.

II. 2. 2. Méthode asynchrone

Cette méthode introduite dans [50] a pour but de réduire le spectre électromagnétique des circuits asynchrones. En effet, nous avons vu que les circuits asynchrones rayonnent moins que les circuits synchrones par construction. Cependant, il est nécessaire de réduire encore plus leur rayonnement pour respecter les normes. Dans ce but, une méthode de mise en forme du courant est proposée avec quatre étapes majeures [50], [51].

La première étape de cette méthode est la modélisation de l'architecture du circuit. Pour cela, un graphe de flot de contrôle de données (*Control Data Flow Graphs, CDFG*) est utilisé. Dans cette modélisation, chaque opération est représentée par un cercle et les liaisons entre les ressources sont représentées par un arc, comme montré sur la Figure II.5 où l'architecture d'un circuit est représentée (a) avec son graphe (b). Cette représentation permet d'identifier le séquençement, le parallélisme et les choix des données dans le circuit. Par exemple, dans la Figure II.5, les opérations OP2, OP3 et OP4 sont concurrentes.

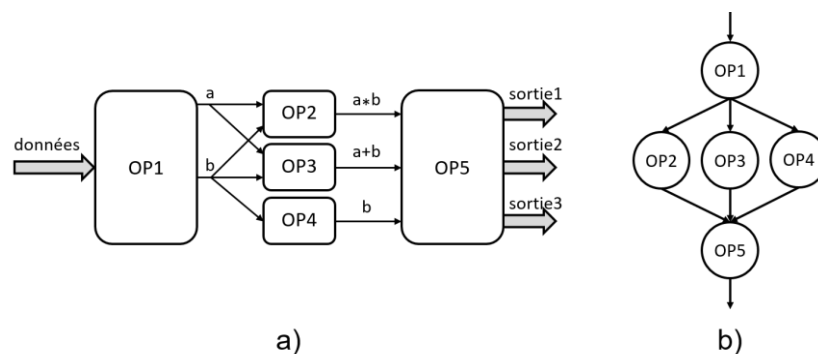


Figure II.5. Exemple a) d'architecture d'un circuit et b) de son CDFG.

Une fois la modélisation du circuit obtenue il est nécessaire d'extraire le délai de chaque opération. Cette deuxième étape est réalisée en effectuant une synthèse logique du circuit et en étudiant le chemin critique des opérations. Les délais extraits de la synthèse permettent d'annoter le graphe.

Durant la troisième étape, le profil du courant est modélisé pour chaque opération. L'activité en courant des opérations dépend du protocole de communication utilisé dans le circuit. Pour déterminer la modélisation d'un protocole celui-ci est décomposé en différentes phases. Chaque phase représente des actions majeures qui composent le protocole. Par exemple, le protocole quatre phases décrit dans le chapitre I, peut être décomposé en deux phases. La première correspond à l'arrivée de la donnée valide avec sa requête. La deuxième représente l'ouverture et la fermeture du registre ainsi que la fin du protocole, avec la fin de la requête et de l'acquiescement. La consommation de chaque phase peut être modélisée par un triangle, comme représenté sur la Figure II.6 pour le protocole quatre phases. Ainsi, dans

le cas de ce protocole, chaque place du graphe du flot de contrôle de données peut être remplacée par deux places pour représenter les deux phases.

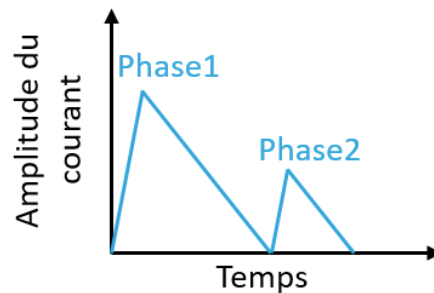


Figure II.6. Représentation des deux phases de courant du protocole quatre phases.

La dernière étape permet de répartir la consommation de courant dans le temps. Pour cela, les opérations qui ne se trouvent pas sur le chemin critique sont décalées dans le temps. Cela permet de décaler les appels de courant sans augmenter la latence du circuit. Reprenons l'exemple de la Figure II.5, où les opérations OP2, OP3 et OP4 sont concurrentes. La Figure II.7, nous permet de constater que l'opération OP2 est sur le chemin critique mais OP3 et OP4 ne le sont pas. Il nous reste donc du temps pour décaler les opérations OP3 et OP4 de respectivement deux et trois unités de temps. Nous pouvons ainsi répartir les opérations dans le temps, de manière à minimiser la consommation de courant simultanée sans augmenter la latence du circuit. Cette opération est réalisée grâce à un algorithme d'ordonnancement. Pour chaque opération du système une « force » est calculée afin de représenter l'impact de l'affectation de l'opération à une étape donnée. Ensuite, l'algorithme ordonne les opérations en minimisant les forces sans augmenter la latence. Cet algorithme est appelé *Force Directed Scheduling* [52].

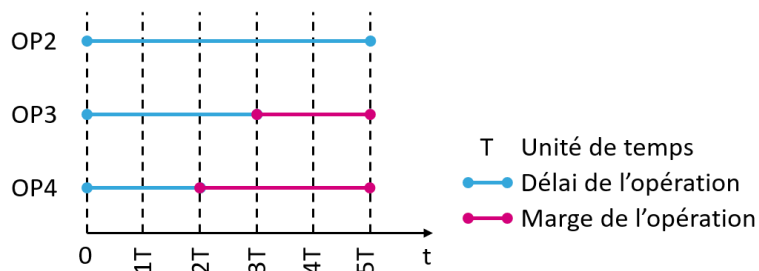


Figure II.7. Illustration des délais des opérations et de la marge pour les décaler.

Cette méthode a été appliquée à un filtre asynchrone à réponse impulsionnelle finie, avec quatre coefficients, implémenté en technologie 0,18 μm STMicroelectronics. La Figure II.8 montre le spectre du courant du filtre avant et après l'utilisation de la méthode de mise en forme du courant. Un gain sur l'amplitude maximale de 9 dB et de 1 dB pour les fréquences au-delà de 500 MHz est obtenu. Ces résultats peuvent être améliorés en appliquant cette méthode sur un circuit qui possède plus de concurrentes entre les opérations. En effet, seulement deux opérations sont concurrentes dans ce filtre [51].

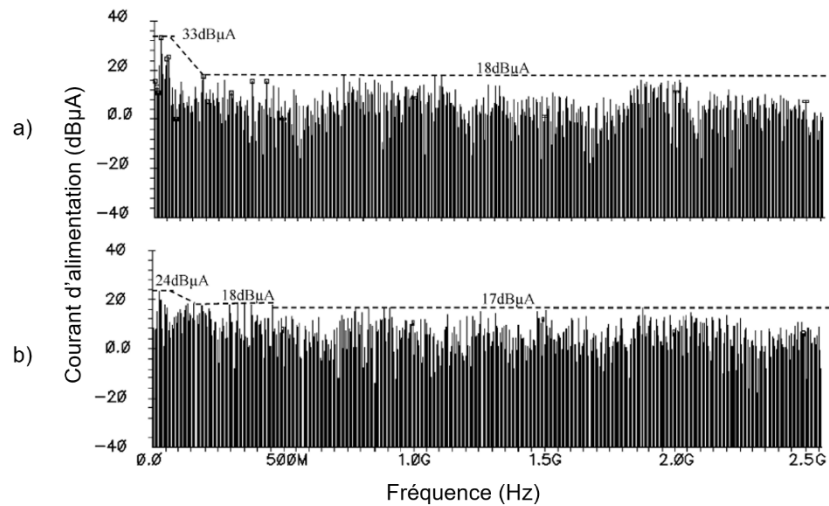


Figure II.8. Spectre du courant du filtre asynchrone a) avant et b) après l'utilisation de la méthode de contrôle du spectre [50].

Cette méthode de conception permet de répartir la consommation de courant afin de réduire le spectre électromagnétique des circuits numériques asynchrones. Les opérations du système sont réparties, grâce à un algorithme, de manière à répartir les opérations concurrentes sans augmenter la latence du circuit. Cependant, elle ne permet pas de contrôler le spectre afin d'éviter une plage de fréquences précise. C'est pourquoi une nouvelle méthode de conception est nécessaire.

II. 3. Introduction d'un gabarit

Il existe une méthode de conception en synchrone qui permet de réduire une plage de fréquences mais qui ne permet pas de définir le niveau maximum des émissions tolérées pour chacune des fréquences. Il existe une méthode asynchrone, qui permet de réduire le rayonnement mais pas de le contrôler. Or, pour pouvoir certifier un produit et le commercialiser, ce dernier doit respecter un niveau de rayonnement pour chaque fréquence spécifiée par les normes CEM. C'est pourquoi, il est nécessaire d'avoir un gabarit en fréquences défini dès l'étape de conception du circuit.

II. 3. 1. Principe

La première étape, pour définir un gabarit fréquentiel est de se renseigner sur les normes à respecter suivant l'application finale du circuit. Ces normes sont spécifiques au domaine d'application du système, nous comprenons bien que les contraintes ne sont pas les mêmes dans le domaine médical ou dans le domaine automobile. Une fois la norme sur les émissions électromagnétiques identifiée, nous obtenons directement un premier gabarit à respecter.

Cependant, suivant le cahier des charges et les objectifs du circuit, le concepteur peut vouloir contraindre les émissions électromagnétiques de son circuit de manière plus agressive. Il est alors possible pour le concepteur de modifier le gabarit initial pour diminuer le niveau des amplitudes tolérées pour certaines fréquences.

Ce gabarit est ajouté au cahier des charges du circuit et le flot de conception doit assurer que le rayonnement du circuit respecte cette nouvelle contrainte. L'ajout de celle-ci permet ainsi d'éviter d'avoir un retour du produit pour non-respect des normes électromagnétiques ce qui obligerait à faire des modifications sur le circuit. De plus, l'ajout de contraintes par le concepteur peut permettre d'éviter un dysfonctionnement du circuit, surtout lorsqu'il s'agit d'un circuit mixte, avec une partie analogique et une partie numérique. Il est connu que dans ces systèmes la partie numérique a tendance à perturber la partie analogique, qui est par nature beaucoup plus sensible aux bruits parasites [53].

II. 3. 2. Deux Exemples de Normes

II. 3. 2. 1. Communication Ultra-Large Bande

Les circuits pour les communications Ultra-Large Bandes (*Ultra Wide Band UWB*) aux Etats-Unis doivent respecter le gabarit spectral défini par la Commission Fédérale des Communications (Figure II.9) [29].

La technique de modulation radio ultra-large bandes est basée sur la transmission de données à travers des impulsions de très courtes durées, souvent inférieures à la nanoseconde. Le spectre de ces transmissions a donc une très grande bande passante. L'amplitude du spectre d'un circuit UWB doit au minimum respecter le gabarit présenté sur la Figure II.9.

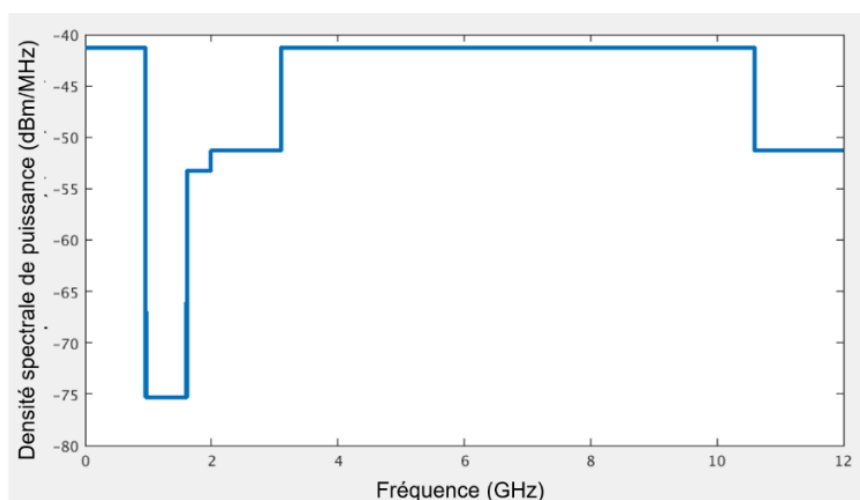


Figure II.9. Gabarit spectral pour la compatibilité électromagnétique de la Commission Fédérale des Communications (FCC).

Il existe également des normes pour ces communications en Europe. Elles sont définies par l'Institut Européen des Normes de Télécommunications (*European Telecommunications Standards Institute ETSI*) [54].

II. 3. 2. 2. Communication en Champ Proche

La communication en champ proche est une technique de communication sans fil mais à courte distance. En effet, les deux systèmes doivent être à une distance de quelques centimètres pour pouvoir échanger des informations. Cette méthode de communication est très utilisée pour les paiements sans contact chez les commerçants ou pour valider son titre de transport en commun.

Les informations sont échangées entre les deux systèmes à travers un champ magnétique émis par le lecteur. Il vient moduler en amplitude son champ pour transmettre les informations à la carte. Ensuite, pour répondre, la carte vient charger le champ avec un signal de sous-porteuse (modulation de charge) pour transmettre les données [55], comme montré sur la Figure II.10.

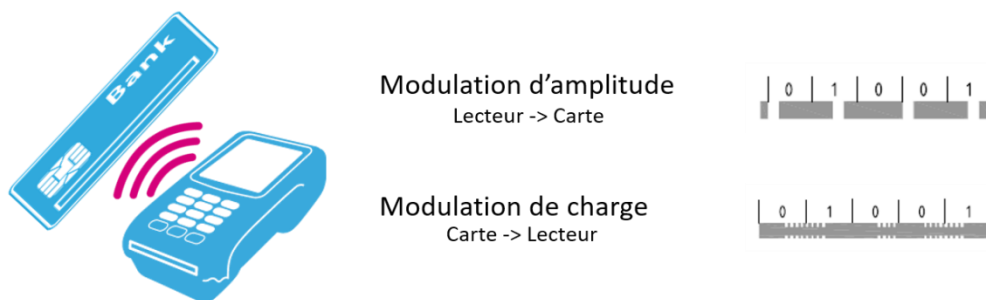


Figure II.10. Exemple des modulations du protocole de Communication à Champ Proche entre une carte et un lecteur [55].

Les deux modulations de cette communication sont réglementées par des standards internationaux ISO/IEC [55]–[58]. Une fois que le lecteur a envoyé une commande, il se met en attente d'une réponse de la carte. Or, avant de répondre la carte doit analyser la demande et générer sa réponse. Pendant cette phase de traitement des données le courant dynamique consommé peut engendrer un effet de modulation de charge sur le champ magnétique [57]. Dans certains cas, le lecteur va interpréter ces changements comme une information non valide émise par la carte. Il y a alors une erreur de protocole. Le lecteur peut penser qu'il est attaqué par un dispositif malveillant. Pour se protéger il peut aller jusqu'à détruire la carte. C'est pourquoi, il est très important de respecter les limites des émissions électromagnétiques indiquées dans les normes ISO/IEC 14443-2 [55] et 14443-3 [57].

II. 4. Présentation de notre méthode

Nous avons vu au travers des deux exemples précédents que les circuits doivent respecter des niveaux d'émissions pour chaque fréquence. Si cette contrainte n'est pas prise en compte dès l'étape de conception, le circuit risque d'émettre en dehors des normes. Pour pouvoir être commercialisé, des changements devront être effectués pour réduire son rayonnement. Si les spécifications des normes sont prises en compte dès l'étape de conception, du temps sera gagné sur la commercialisation du circuit. Notre méthode a pour but de prendre en compte un gabarit fréquentiel et de concevoir un circuit dont les émissions respecteront les contraintes imposées par ce gabarit.

L'objectif de notre méthode est de placer les appels de courant des différents étages du circuit à des instants qui permettent d'avoir un spectre en dessous du gabarit. Un délai supplémentaire est déterminé et ajouté sur le chemin de la requête de manière à mémoriser une donnée valide sans émettre un rayonnement perturbateur.

Pour cela, nous avons choisi d'utiliser des circuits asynchrones pour bénéficier de leurs avantages en termes d'émission et de suppression des harmoniques liées au signal d'horloge. Pour mettre en œuvre notre méthode nous avons ciblé les circuits micropipelines. En effet, ils ont l'avantage d'avoir une conception très proche de celle des circuits synchrones. Leur architecture est donc plus facilement comprise par des concepteurs avec une formation synchrone.

II. 4. 1. Hypothèse pour l'émission électromagnétique

Pour pouvoir concevoir notre circuit en respectant un gabarit fréquentiel nous avons besoin d'estimer le rayonnement du circuit. Pour cela, nous avons fait l'hypothèse que le spectre électromagnétique est directement lié à la consommation de courant du circuit. Les appels de courant du circuit sur l'alimentation vont engendrer les émissions électromagnétiques. En effet, le champ électromagnétique est le résultat de l'évolution du courant et de la tension dans le temps (chapitre I.3.1).

Nous avons décidé, dans un premier temps, de ne pas prendre en compte la manière de réaliser le placement et le routage du circuit. En effet, le routage, des fils du circuit et des alimentations, peut avoir une influence sur le rayonnement. Néanmoins, nous considérons que ces paramètres ont une influence du second ordre sur le rayonnement électromagnétique par rapport aux appels de courant du circuit.

Ainsi, pour obtenir le spectre du circuit, une transformée de Fourier (*Fast Fourier Transform FFT*) est réalisée sur la courbe de la consommation de courant. La Figure II.11 montre le résultat, en simulation, de l'application de la transformée de Fourier sur une modélisation de la consommation en courant en synchrone et en asynchrone.

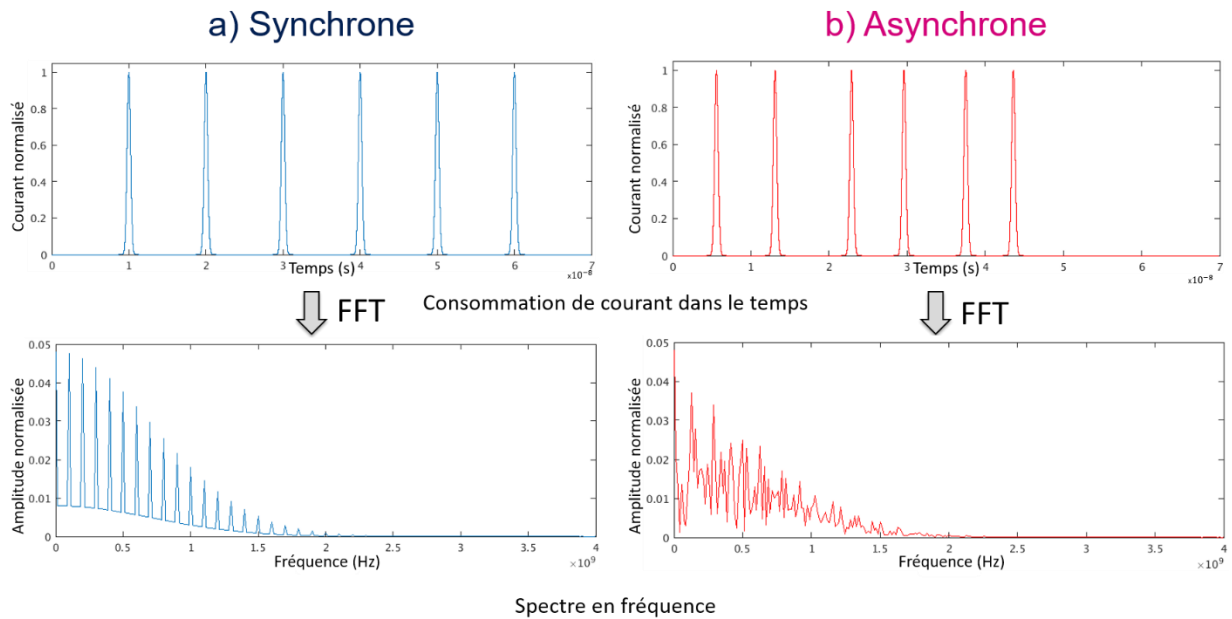


Figure II.11. Exemple en simulations de l'utilisation de la transformée de Fourier sur la modélisation de la consommation de courant d'un circuit a) synchrone et b) asynchrone pour obtenir le spectre électromagnétique.

D'après notre hypothèse, un changement sur la consommation de courant va engendrer un changement sur le spectre électromagnétique du circuit. C'est pourquoi, pour optimiser le rayonnement des circuits asynchrones, nous allons mettre en forme la consommation de courant dans le temps en décalant les appels de courant du circuit.

II. 4. 2. Ajout de délais

Dans un circuit asynchrone, seuls les étages qui ont des données valides à traiter vont fonctionner et donc consommer du courant. Seul un étage qui a une requête, présente en entrée, est activé. Cette propriété répartit naturellement les pics de courant dans le temps.

La Figure II.12.a représente un circuit micropipeline à 4 étages. A chaque étage un délai est inséré sur le chemin de la requête pour couvrir le temps de traitement de la logique combinatoire associée. Ce délai permet de mémoriser une donnée valide dans les registres. Lorsqu'une requête est envoyée dans le circuit, le premier étage va consommer du courant après un délai d_0 , puis l'étage suivant après un délai d_1 et ainsi de suite. Chaque délai a une valeur qui lui est propre en fonction de sa logique combinatoire. D'après ces valeurs, nous pouvons en déduire les différents instants où chaque étage du circuit va consommer du courant. Un pic de courant peut alors être associé à chaque instant de commutation. Ainsi, nous obtenons le modèle de la consommation du circuit.

La Figure II.12.b montre la courbe en courant du circuit de la Figure II.12.a. Pour simplifier la représentation, les pics en courant sont modélisés grâce à une fonction triangle. Ils ont tous

les mêmes paramètres, ce qui ne sera pas le cas par la suite. Notre modèle de pic de consommation de courant sera présenté dans le chapitre III.

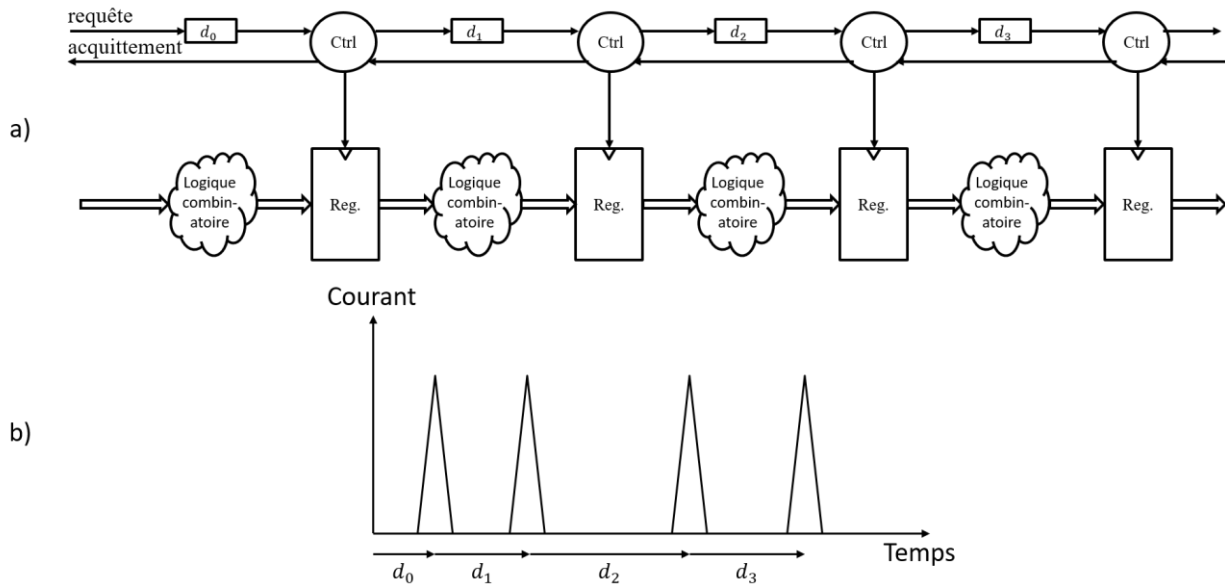


Figure II.12. a) Micropipeline de quatre étages et b) sa consommation en courant en simplifiant les pics de courant par des triangles tous identiques.

Notre objectif est de déplacer les pics de courant de manière à obtenir un spectre électromagnétique en dessous du gabarit spectral. Pour bouger les pics en courant, un délai est ajouté, sur le chemin de la requête, en plus du délai minimum nécessaire. Ajouter un retard supplémentaire à la requête ne va pas modifier le résultat du traitement de la donnée, il sera seulement mémorisé plus tard.

La Figure II.13.a montre l'architecture d'un étage micropipeline modifié avec l'ajout du délai sur le chemin de la requête. La Figure II.13.b illustre l'impact de cet ajout sur la consommation de courant de cet étage, le pic en courant de départ, en pointillés, est décalé de la valeur $d_{ajouté}$ dans le temps, le pic en bleu sur la figure.

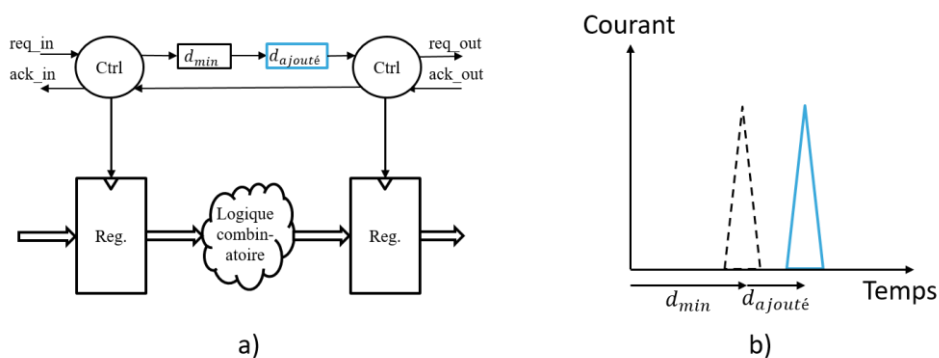


Figure II.13. a) Architecture d'un étage micropipeline avec un délai ajouté et b) sa consommation de courant avec le déplacement du pic.

D'après notre hypothèse, présentée dans la partie précédente, l'ajout de ces délais dans le circuit micropipeline nous permet de déplacer les pics de consommation de courant dans le

temps et de changer le spectre du circuit. La simulation réalisée sur la Figure II.14 montre les courbes dans les domaines temporels et fréquentiels pour deux circuits micropipelines à quatre étages. Le premier, dont les courbes sont en bleues, est le circuit de référence, avec des valeurs de délais données Équation II-1.

$$\begin{cases} d_0 = 2 \text{ ns} \\ d_1 = 3 \text{ ns} \\ d_2 = 4 \text{ ns} \\ d_3 = 3 \text{ ns} \end{cases} \quad \text{Équation II-1}$$

Dans le deuxième circuit, dont les courbes sont rouges, des délais supplémentaires de 1 ns ont été ajoutés aux étages un et deux. La Figure II.14.a représente la simulation des courbes en courant des deux circuits avec le décalage dans le temps de la courbe rouge. La Figure II.14.b montre les spectres après la transformée de Fourier appliquée sur le courant. Nous constatons un changement entre les deux spectres dû aux délais ajoutés dans le deuxième circuit.

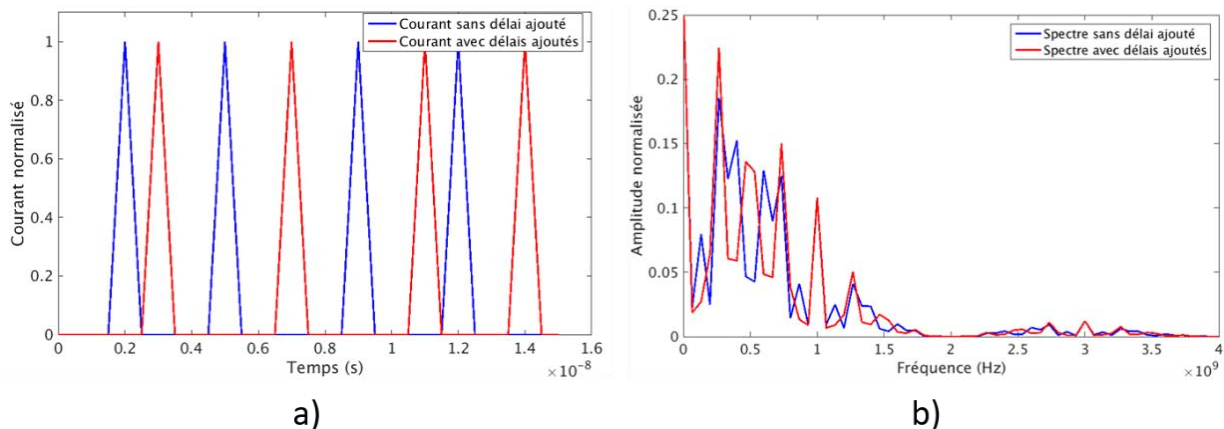


Figure II.14.a) Consommation de courant pour un circuit micropipeline de quatre étages sans ajouter de délai (courbe bleu) et en ajoutant du délai (courbe rouge) et b) les spectres associés.

Sur la Figure II.14.b, le spectre rouge, correspondant au circuit avec les délais ajoutés, n'est pas optimisé pour toutes les fréquences par rapport au spectre bleu du circuit de référence. Il sera donc important de déterminer les bonnes valeurs de délais pour avoir une optimisation sur les plages de fréquences déterminées par le gabarit.

II. 4. 3. Détermination des valeurs de délais

L'objectif est maintenant de trouver les valeurs pour les délais supplémentaires du circuit afin que le rayonnement respecte le gabarit défini par le concepteur. Pour cela, une première approche analytique est présentée, suivie d'une approche algorithmique.

II. 4. 3. 1. Approche analytique

L'approche analytique doit nous permettre de trouver les valeurs de délais à partir des limites en fréquences définies par le gabarit.

Pour simplifier les calculs, et notamment le calcul de la transformée de Fourier, les appels de courant sont modélisés par des Dirac. Ainsi, la courbe de consommation de courant d'un circuit peut être représentée par un peigne de Dirac pour la propagation d'une donnée et ce peigne va se répéter à la donnée suivante, comme le montre la Figure II.15. L'espacement entre deux pics consécutifs est la valeur du délai sur le chemin de la requête. En revanche, l'écart entre l'origine et un pic correspond à la somme des délais sur le chemin de la requête depuis l'entrée du circuit. Notons d_n le délai entre l'étage $n-1$ et n , et D_n le délai entre l'étage n et l'entrée du circuit, nous obtenons alors l'Équation II-2, cette notation est utilisée sur la Figure II.15.

$$D_n = \sum_{p=0}^n d_p \text{ avec } D_0 = d_0 \tag{Équation II-2}$$

De plus, chaque délai sur le chemin de la requête est borné (Équation II-3). Sa valeur minimale est déterminée par le chemin critique de la logique combinatoire qui lui est associée. Sa valeur maximale est définie par le concepteur pour pouvoir borner la latence de chaque étage et ne pas avoir un circuit trop lent. La somme de tous les délais du chemin de contrôle, dans un circuit linéaire, donne la latence de la propagation d'une donnée dans le circuit. Cette valeur de latence est, en général, contrainte pour avoir de bonnes performances pour le circuit. Elle peut ensuite être répartie entre tous les étages à condition de respecter la valeur minimale définie par la logique combinatoire.

$$d_{n,min} \leq d_n \leq d_{n,max} \tag{Équation II-3}$$

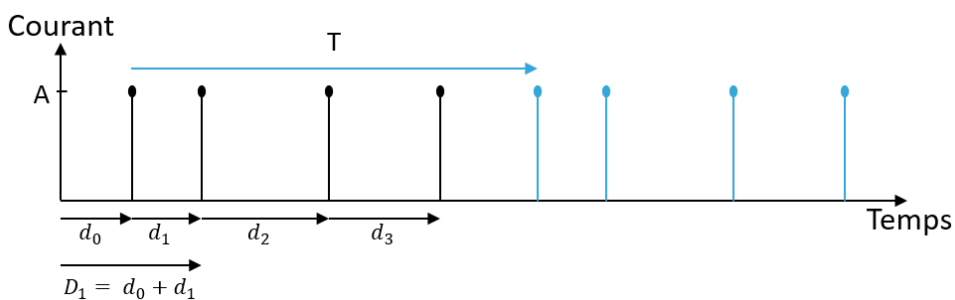


Figure II.15. Représentation de la consommation de courant avec un peigne de Dirac pour la propagation de deux données. T est la période entre deux données, A l'amplitude du courant, d_n le délai entre l'étage $n-1$ et n et D_n le délai entre l'étage n et l'entrée du circuit.

Nous pouvons expliciter la formule de la consommation de courant comme dans l'Équation II-4 :

$$i(t) = \sum_{p=0}^n \left(\sum_k A_p \delta(t - kT - D_p) \right) \quad \text{Équation II-4}$$

où $k \in \mathbb{R}$, A_p est l'amplitude du Dirac de l'étage p , chaque pic peut avoir une hauteur différente, D_p est la combinaison linéaire des délais qui précèdent l'étage p et n est le nombre d'étages du circuit micropipeline. En appliquant la transformée de Fourier sur la consommation de courant nous obtenons la fonction $I(f)$ explicitée sur l'Équation II-5.

$$F[i(t)] = I(f) = \sum_{p=1}^n \left(\sum_k \frac{A_p}{T_0} \delta(f - kf_0) e^{-2i\pi D_p f} \right) \quad \text{Équation II-5}$$

Nous constatons que les délais introduisent un facteur de phase dans le spectre. En inversant l'ordre des sommes nous obtenons l'Équation II-6.

$$I(f) = \sum_k \sum_{p=1}^n \frac{A_p}{T_0} * e^{-2i\pi D_p f} * \delta(f - kf_0) = \sum_k G(f) \delta(f - kf_0) \quad \text{Équation II-6}$$

Où

$$G(f) = \sum_{p=1}^n \frac{A_p}{T_0} e^{-2i\pi D_p f} \quad \text{Équation II-7}$$

Dans l'Équation II-7 de $G(f)$, f peut être remplacé par kf_0 , avec $k \in \mathbb{R}$, car pour les autres valeurs de f la fonction $I(f)$ est nulle d'après l'Équation II-6. Nous souhaitons que le module de $G(f)$ soit inférieur au gabarit.

$$|G(kf_0)| < \text{Gabarit}(kf_0) \quad \text{Équation II-8}$$

$$\left| \sum_{p=1}^n \frac{A_p}{T_0} e^{-2i\pi D_p kf_0} \right| < \text{Gabarit}(kf_0) \quad \text{Équation II-9}$$

Avec l'Équation II-8 et l'Équation II-9 nous obtenons une majoration du module de $G(kf_0)$. Cependant, nous n'avons pas réussi à résoudre l'inéquation et à en déduire les valeurs des délais à insérer pour respecter le gabarit défini. C'est pourquoi, nous avons décidé d'utiliser une approche algorithmique, qui va nous permettre de trouver un résultat plus rapidement.

II. 4. 3. 2. Approche algorithmique

L'approche algorithmique permet de trouver une solution à notre problème en un temps raisonnable. L'objectif est toujours d'optimiser le spectre électromagnétique du circuit de manière à respecter le gabarit défini par les normes ou par le concepteur. Pour cela, nous avons décidé d'utiliser un algorithme génétique [59].

Cet algorithme est inspiré de la biologie et de phénomènes naturels [59]. Une sélection naturelle est réalisée sur une population d'individus. L'algorithme manipule une population composée d'individus, qui sont chacun une solution potentielle. Comme en biologie, chaque individu est composé de gènes, qui sont ici les paramètres du problème. Des processus biologiques sont également utilisés dans les différentes étapes de l'algorithme [59].

L'algorithme génétique commence avec une population initiale composée de N individus, tous candidats pour résoudre le problème. Chaque individu est composé de tous les paramètres du problème. Ensuite, chaque individu x est évalué grâce à une fonction de coût $f(x)$ qui permet de déterminer s'il est une bonne solution pour résoudre le problème. Lors de cette étape, la fonction attribue une note à chaque individu. En général, l'objectif est d'optimiser une fonction, donc la note la plus proche de zéro sera la meilleure. Dans l'étape suivante, une nouvelle génération est créée.

Dans un premier temps, une sélection est effectuée dans la population courante pour choisir les parents de la génération suivante. Pour cela, chaque individu est représenté sur une ligne avec une section de longueur inversement proportionnelle à sa note. Ensuite, des individus sont sélectionnés de façon proportionnelle sur cette ligne. Cette méthode permet de privilégier les individus qui s'adaptent le mieux au problème [60].

Ensuite, la population suivante est générée à partir de ces parents en trois étapes. Dans la première étape, les parents qui ont la meilleure note sont directement réinjectés dans la prochaine génération. L'élite de la population est conservée. En effet, nous ne pouvons pas garantir que les enfants, créés dans les prochaines étapes, seront meilleurs que les parents. Il faut donc garder les meilleures solutions. L'étape suivante est la reproduction, comme en biologie, deux parents sont sélectionnés et une combinaison de leurs gènes est effectuée. Pour cela, un vecteur binaire aléatoire, de taille égale au nombre de gènes d'un individu, est créé. Lorsque la valeur du vecteur vaut 1 le gène du premier parent est sélectionné, sinon le gène du deuxième parent est gardé. Enfin, la dernière étape est appelée mutation. Des changements aléatoires sont effectués sur certains individus. La mutation permet de créer de nouvelles solutions pour éviter de converger dans une direction qui ne donnera peut-être pas de solution.

Ainsi, une nouvelle génération d'individus est créée avec de nouveaux candidats pour résoudre le problème. Cette population est alors évaluée avec la fonction de coût $f(x)$. Si une solution est trouvée l'algorithme s'arrête, sinon une nouvelle génération est à nouveau créée

et le processus est répété. Ces étapes de l'algorithme sont résumées sous forme de schéma sur la Figure II.16.

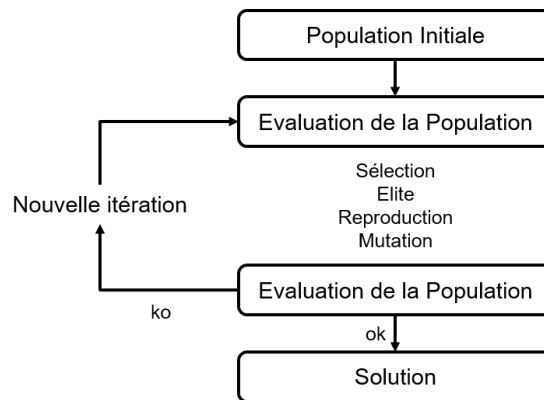


Figure II.16. Présentation de l'algorithme génétique.

Notre objectif est de trouver les valeurs des délais du circuit de manière à respecter le gabarit fréquentiel pour le rayonnement électromagnétique. Pour cela, nous allons créer une population de cinquante individus dont les gènes correspondent aux délais à ajouter dans le circuit. Chaque gène est donc borné par une valeur minimale en fonction de la logique combinatoire et par une valeur maximale définie par les spécifications du circuit, de la même manière que pour l'approche analytique. L'objectif de l'algorithme est de trouver une combinaison de délais permettant de respecter le gabarit. La consommation de courant est extraite à partir des gènes de l'individu grâce à une méthode expliquée dans le chapitre III. A partir de la courbe en courant, la fonction de coût va évaluer le spectre de l'individu, grâce à une transformée de Fourier. L'amplitude du rayonnement sera comparée, pour toutes les fréquences, avec la limite du gabarit. La différence entre le spectre et le gabarit, lorsqu'elle est positive (spectre au-dessus), est ajoutée à la note de l'individu. Le but est de trouver un individu avec une note de zéro, signifiant que tout le spectre est en dessous du gabarit.

Cette approche algorithmique permet de trouver un jeu de délais pour implémenter notre circuit et avoir un rayonnement qui respecte les contraintes des normes et du concepteur. Il n'y a pas de recherche de la « meilleure » solution. En effet, nous avons seulement besoin de respecter le gabarit et plusieurs combinaisons de délais peuvent permettre de le respecter. Nous avons seulement besoin d'en trouver une. Si un rayonnement plus faible est souhaité, il faut redéfinir le gabarit pour avoir des contraintes plus agressives.

II. 5. Conclusion

Dans la première partie de ce chapitre, deux méthodes de contrôle et de réduction du spectre électromagnétique des circuits numériques ont été présentées. La première méthode cible des circuits synchrones. Elle a pour but de réduire le rayonnement sur une plage définie par l'utilisateur ainsi qu'à la fréquence fondamentale du signal d'horloge. Cependant, elle ne permet pas à l'utilisateur de définir une amplitude maximale pour les émissions. De plus, cette

méthode peut être limitée par les contraintes temporelles du circuit à cause du délai à ajouter sur le signal d'horloge. La deuxième méthode concerne les circuits asynchrones. Elle a pour but de réduire le rayonnement électromagnétique en optimisant l'ordonnancement des opérations du circuit. Néanmoins, avec cette méthode le rayonnement n'est pas contrôlé mais seulement réduit. Or, pour certifier un circuit et le commercialiser, il est nécessaire de respecter certaines amplitudes d'émission.

L'introduction d'un gabarit en fréquences, dans la deuxième partie, permet de définir les limitations en amplitude du spectre électromagnétique. En effet, il existe déjà, pour chaque application de circuits, des normes qui définissent les émissions tolérées pour chaque fréquence. Cependant, pour des raisons de spécification ou de différenciation d'un circuit, le concepteur peut vouloir contraindre de manière plus agressive le rayonnement de son circuit. Pour cela, à partir du gabarit de la norme, il peut en redéfinir un nouveau avec des amplitudes plus basses pour certaines fréquences.

La dernière partie de ce chapitre présente le principe de notre méthode de contrôle du rayonnement électromagnétique qui utilise un gabarit en fréquences. Nous faisons l'hypothèse que le spectre du circuit est directement lié aux appels de courant sur l'alimentation. Nous considérons que l'évolution du courant et de la tension du circuit dans le temps va provoquer les émissions électromagnétiques. C'est pourquoi, notre objectif est de répartir dans le temps les appels de courant des éléments du circuit pour obtenir un spectre qui respecte le gabarit. Pour cela, un délai supplémentaire est ajouté sur le chemin de la requête pour changer l'instant de commutation des éléments des différents étages. N'ayant pas trouvé de formule théorique pour calculer les valeurs des délais à insérer, nous avons décidé d'utiliser une approche algorithmique. Un algorithme génétique a été mis en place pour trouver un jeu de délais qui convient. Ainsi, notre méthode permet d'utiliser directement un gabarit pour concevoir un circuit.

L'algorithme génétique a besoin d'une estimation de la courbe en courant pour pouvoir évaluer les différentes combinaisons de délais et déterminer si une solution est trouvée. Une transformée de Fourier est appliquée sur la courbe de consommation de courant pour obtenir le spectre électromagnétique et le comparer avec le gabarit. Nous détaillerons dans le chapitre suivant notre méthode de modélisation de haut niveau du courant.

Chapitre III.

Modélisation de haut niveau du courant

III. 1. Introduction.....	48
III. 2. Réseau de Petri temporel	48
III. 2. 1. Principe	48
III. 2. 2. Utilisation.....	50
III. 2. 3. Instanciation	51
III. 3. Estimation du courant.....	54
III. 3. 1. Modèle d'un pic de courant.....	55
III. 3. 2. Modèle de la consommation de courant.....	59
III. 4. Conclusion	60

Résumé :

Pour pouvoir évaluer le spectre électromagnétique du circuit, nous avons vu qu'il est nécessaire de disposer d'un modèle de la consommation de courant. L'obtention de cette courbe nécessite deux étapes.

Premièrement, les réseaux de Petri temporels sont utilisés pour modéliser le chemin de contrôle du circuit. Ce modèle permet au concepteur d'analyser le chemin de contrôle afin de trouver de potentiels problèmes d'inter-blocages et de simuler rapidement le fonctionnement de son circuit. Les délais à respecter avant la transmission des données sont précisés dans le réseau. La simulation du réseau de Petri temporel permet d'extraire les instants de commutation de chacun des éléments de mémorisation du circuit.

Deuxièmement, un modèle utilisant deux courbes exponentielles dos à dos est utilisé pour représenter les pics de courant des étages. La première exponentielle représente la consommation de la propagation de la requête vers les registres qu'elle contrôle et la deuxième la consommation de la logique combinatoire après les registres. Les paramètres des pics sont calculés pour chaque étage.

L'association des temps de commutation avec les pics de chaque étage nous donne un simulateur analogique rapide, qui permet d'obtenir un modèle de la courbe de la consommation de courant totale du circuit.

III. 1. Introduction

Nous avons vu dans le chapitre précédent qu'il est nécessaire de connaître le profil de consommation de courant pour pouvoir estimer le rayonnement électromagnétique d'un circuit. Ce chapitre a pour objectif de présenter notre méthode de modélisation de haut niveau de la consommation de courant des circuits numériques micropipelines.

La première partie introduit tout d'abord la notion de réseaux de Petri. Généralement, ces réseaux servent à représenter de manière graphique le réseau de contrôle des circuits asynchrones. Ils permettent d'aider le concepteur à concevoir son réseau de contrôle et à s'assurer qu'il ne présente pas d'inter-blocages, c'est-à-dire qu'une donnée pourra toujours se propager dans le circuit. Nous verrons qu'il est possible d'enrichir ce réseau de Petri avec un paramètre temporel qui ajoute une contrainte sur le passage des ressources d'un étage à l'autre. Nous utiliserons ces réseaux à diverses fins qui seront présentées dans la suite du document. Dans notre méthode, ils nous permettent également de simuler le circuit de contrôle avec des valeurs pour les délais entre les contrôleurs et ainsi de récupérer les temps de commutation des différents étages. Enfin, nous présenterons notre méthode pour implémenter les réseaux de Petri temporels, basée sur un langage de modélisation et de vérification de circuits, le SystemVerilog. Ce langage a été choisi pour faciliter la réutilisation des outils du monde synchrone et aider à son adoption par les concepteurs.

La deuxième partie présente la modélisation de haut niveau de la consommation de courant du circuit micropipeline et plus particulièrement un nouveau modèle pour un pic de courant dans un étage micropipeline. Ce nouveau modèle, composé de deux exponentielles dos à dos, nous permet d'être proche de la réalité tout en restant relativement simple. Les paramètres des pics de courant sont extraits d'une analyse temporelle statique. Cela nous permet d'avoir un pic de courant spécifique à chaque étage du circuit. Enfin, la modélisation de la consommation de courant totale du circuit est présentée. Pour cela, le pic de consommation de courant est associé à l'instant de commutation de son étage.

III. 2. Réseau de Petri temporel

L'activité dans les circuits asynchrones est concurrente. Tous les éléments évoluent dans le temps séparément tout en communiquant les uns avec les autres. Nous allons voir dans cette partie, qu'il est possible de modéliser le circuit de contrôle en utilisant un modèle de réseau de Petri [61] dans lequel nous ajouterons un paramètre temporel.

III. 2. 1. Principe

Les réseaux de Petri sont basés sur une représentation graphique et non sur un langage [61]. Ils permettent de modéliser des actions qui sont soumises à des conditions

d'activations. C'est graphes sont orientés avec des flèches et ils sont construits avec une alternance de nœuds appelés « places » et de transitions comme le montre la Figure III.1. L'orientation du graphe permet de définir les places d'entrée et de sortie d'une transition. Par exemple dans la Figure III.1, la place notée P_0 est une entrée de la transition T_0 alors que P_1 est une place de sortie. Les places peuvent contenir des jetons qui représentent des ressources disponibles, les transitions permettent de déplacer les ressources d'une place à l'autre selon certaines règles d'activations. En effet, une transition peut s'activer seulement si toutes ses places en entrée possèdent au moins un jeton. Lorsque cette condition est remplie, un jeton est retiré de chacune des places en entrée et un jeton est ajouté à chacune des places en sortie. Deux contraintes supplémentaires peuvent être ajoutées pour l'activation des transitions. Premièrement, des poids peuvent être indiqués sur les arcs du graphique, ils représentent le nombre de jetons qui doivent être déplacés lors de l'activation de la transition. Celle-ci pourra s'activer seulement s'il y a le bon nombre de jetons disponibles dans la place en entrée. Deuxièmement, il est possible de limiter le nombre de jetons par place. De cette manière, si la place de sortie a déjà atteint sa capacité maximale, la transition ne pourra pas s'activer tant qu'au moins un jeton n'est pas retiré, grâce à l'activation de la transition suivante. Ainsi, les activations des différentes transitions se font de manière concurrente tout en étant dépendantes les unes des autres. Il est donc possible de modéliser le comportement du chemin de contrôle d'un circuit asynchrone en utilisant un réseau de Petri [62].

La Figure III.1 illustre le fonctionnement d'un réseau de Petri. Dans notre cas, chaque place peut contenir un seul jeton, donc les poids des arcs sont de un. D'après les règles énoncées la transition T_1 peut s'activer car la place P_1 possède un jeton et les places P_2 et P_3 sont disponibles. Cependant, la transition T_3 ne peut pas s'activer car sa place d'entrée P_2 est vide. Une fois que la première activation a eu lieu la transition T_3 peut se déclencher car la transition T_1 a retiré un jeton de la place P_1 et en a placé un dans la place P_2 et un dans la place P_3 .

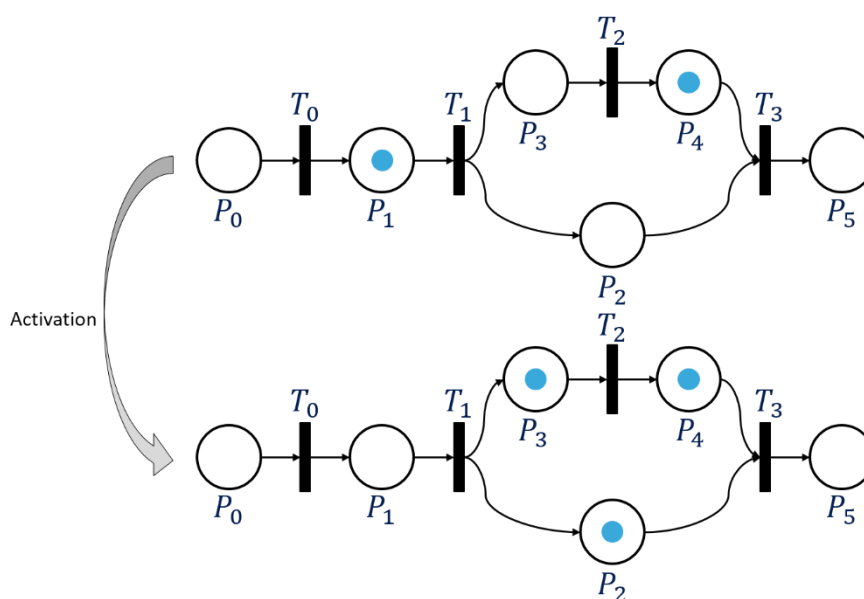


Figure III.1. Illustration d'un réseau de Petri et exemple de son activation.

Il existe des extensions des réseaux de Petri dans lesquelles la notion de temps est introduite [63], [64]. En effet, dans les *Time Petri Net (TPN)* [63] deux valeurs temporelles sont définies pour chaque arc. La première définit le temps minimum qui doit s'écouler avant l'activation de la transition à partir du moment où toutes les places en entrée ont un jeton disponible. La deuxième valeur définit le temps maximum pendant lequel les conditions en entrée de l'activation sont remplies sans que la transition se déclenche. Une fois ce temps écoulé, la transition doit se déclencher et faire passer les jetons sur les sorties. Ainsi, ces deux valeurs donnent un intervalle de temps pendant lequel la transition doit s'activer et transmettre les données [63]. Une deuxième variante des réseaux de Petri temporels est nommée *Timed Petri Net (TdPN)* [64]. Dans ce modèle, chaque transition est associée à une valeur de délai. Une fois que la transition peut s'activer elle doit attendre que le délai spécifié soit écoulé et ensuite elle transmet la ressource à la place suivante.

Notre réseau de Petri temporel se rapproche des *Timed Petri Net*. Nous allons spécifier une valeur de délai pour chaque place. Ce délai représente le temps minimum que doit passer le jeton dans la place. La transition après cette place pourra s'activer seulement une fois que ce délai est écoulé. Cependant, à la différence des *TdPN*, si les autres conditions pour l'activation de la transition ne sont pas réalisées après ce délai, le jeton restera en attente de ces conditions dans la place. L'ajout de ce délai sera détaillé dans les parties suivantes.

III. 2. 2. Utilisation

La recherche des instants de commutation est essentielle pour obtenir la consommation de courant. C'est pourquoi, une modélisation de notre circuit est nécessaire afin qu'une simulation rapide nous permette de récupérer ces informations.

Pour cela, nous avons choisi d'utiliser les réseaux de Petri avec une notion temporelle. Cette modélisation permet de représenter le réseau de contrôle du circuit pour simuler son fonctionnement tout en incluant les informations sur les délais.

Pour obtenir le réseau de Petri d'un circuit, le concepteur peut l'extraire à partir d'un langage de description matérielle de l'architecture (la description *Register Transfer Level, RTL*) du circuit ou depuis un fichier qui liste toutes les entités utilisées et leurs interconnexions (appelé *netlist* en anglais). Les canaux de communication et les éléments de mémorisation sont modélisés par des transitions (les barres verticales de la Figure III.2.b), tandis que les parties de logiques combinatoires et les délais associés sont modélisés par des places (les cercles représentés sur Figure III.2.b).

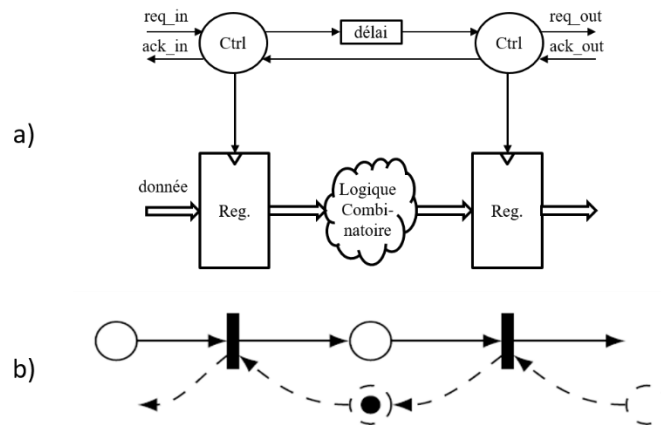


Figure III.2. a) Circuit micropipeline et b) son modèle de réseau de Petri.

Comme le montre la Figure III.2.b, le chemin direct, de requête, est décrit mais le chemin de retour, d'accusé de réception, est également représenté en pointillés. Cette représentation du modèle va également permettre au concepteur d'évaluer l'impact du protocole sur le comportement du circuit. Il permet de simuler plusieurs protocoles et de choisir celui qui convient le mieux pour les spécifications du circuit.

De plus, le modèle permet de représenter les canaux de communication lorsqu'il y a une fourche ou une jonction de plusieurs chemins. Si le chemin de contrôle se sépare en deux parties pour traiter des données en parallèle, le circuit micropipeline est dit non linéaire [65] (comme sur la Figure III.1). Ainsi, le réseau de Petri permet de modéliser et de simuler le fonctionnement du chemin de contrôle du circuit tout en respectant l'ordonnement d'activation des registres.

III. 2. 3. Instanciation

Dans notre méthode, nous avons choisi de décrire notre réseau de Petri temporel en utilisant le langage SystemVerilog. Ce langage, standardisé par l'*Institut des ingénieurs Electriciens et Electroniciens IEEE* [66], est un langage de description et de vérification de matériel, il est utilisé pour modéliser, simuler, tester et implémenter des circuits électroniques. Le SystemVerilog a l'avantage d'utiliser des simulateurs basés sur des événements, ce qui dans notre cas, permet de simuler efficacement les réseaux de Petri temporels.

Les transitions dans le modèle du réseau de Petri temporel représentent les contrôleurs, qui synchronisent le circuit. Le comportement d'une transition est alors remplacé par le comportement d'un contrôleur décrit en SystemVerilog. Le contrôleur est décrit en fonction du protocole utilisé dans le circuit. Ainsi, pour évaluer différents protocoles il suffit de remplacer la description en SystemVerilog de la transition par la description d'un autre protocole.

La Figure III.3 représente deux contrôleurs, un fonctionnant avec un protocole quatre phases (a) et un fonctionnant avec un protocole deux phases (b). Leur description en SystemVerilog est utilisée pour le réseau de Petri temporel pour décrire les transitions. Ils ont tous les deux les mêmes ports d'entrée et de sortie, ils sont donc facilement interchangeables pour évaluer leur impact sur le comportement du circuit. Dans ce travail de thèse, nous utilisons le contrôleur de la Figure III.3.a.

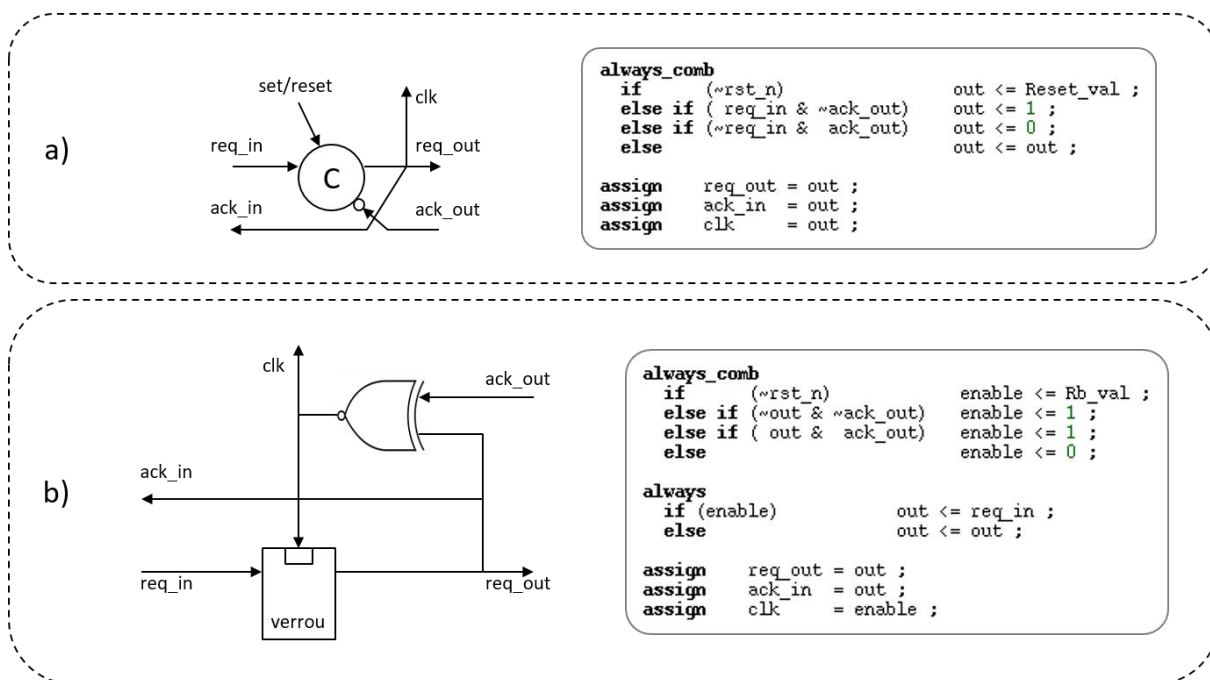


Figure III.3. Représentation d'un contrôleur et sa description en SystemVerilog pour a) un protocole 4 phases réalisé avec une porte de Muller b) un protocole 2 phases réalisé avec un contrôleur MOUSETRAP [67].

Les places dans le réseau de Petri temporel représentent le délai entre deux contrôleurs, c'est-à-dire, au minimum le délai de la logique combinatoire pour mémoriser une donnée valide et au maximum la limite fixée par le concepteur. La valeur minimale est obtenue durant l'analyse temporelle statique du circuit. Au cours de cette étape, les différentes valeurs de délais du circuit sont extraites ce qui permet de déduire le temps minimum nécessaire entre deux contrôleurs. Le délai spécifié pour une place est le temps minimum à respecter à partir du moment où le jeton est présent dans la place. Néanmoins, si une fois ce délai écoulé, les autres conditions d'activation de la transition ne sont pas remplies alors la place continue à attendre le déclenchement de la transition. C'est la somme entre le délai spécifié pour une place et le délai supplémentaire lié à l'attente des autres conditions d'activation que nous souhaitons extraire.

Les temps stockés dans les places sont spécifiés grâce à un fichier standard de délais (*Standard Delay Format SDF*). Ce type de fichier, généré durant l'analyse temporelle statique du circuit, permet de rétro-annoter les simulations fonctionnelles avec les délais des portes du circuit. Pour chaque porte, le délai est précisé pour les différentes conditions de fonctionnement, rapide, typique ou lent (min, typ, max). Ce fichier distingue également le

délai nécessaire pour passer du niveau logique bas vers haut (temps de montée, *rise* en anglais) et celui nécessaire pour la transition du niveau logique haut vers bas (temps de descente, *fall* en anglais). Ainsi, six valeurs de délais différentes sont précisées pour chaque arc. Nous verrons dans le chapitre IV notre méthode de génération du fichier standard de délais utilisé pour simuler notre réseau de Petri temporel.

Un exemple de l'instanciation de notre modèle du réseau de Petri temporel est représenté sur la Figure III.4. Les places représentent le délai entre deux contrôleurs, ce dernier étant spécifié dans un fichier *SDF* qui contient tous les délais du chemin de contrôle du circuit. Le délai pour le temps de montée et de descente du signal de requête est spécifié entre l'instant d'entrée et de sortie du signal dans la place. Ce délai est également précisé pour le signal d'acquittement. La Figure III.4 montre un exemple pour la requête. Les barres verticales représentent les transitions permettant la synchronisation entre deux places. Enfin, les arcs correspondant aux canaux de communication avec le signal de requête sont les arcs pleins tandis que le signal d'acquittement est représenté par les arcs en pointillés. La connexion entre les arcs et les places est réalisée grâce à des interfaces SystemVerilog.

Une transition peut s'activer seulement si trois conditions sont satisfaites :

- Toutes les entrées de la transition doivent avoir un jeton.
- Les temps spécifiés dans les places en entrée sont écoulés.
- Toutes les sorties de la transition doivent être libres, sans jeton.

Comme pour le fonctionnement d'un réseau de Petri classique, lorsqu'une transition se déclenche, un jeton est retiré de chaque place en entrée et un jeton est ajouté à chaque place en sortie de la transition.

Chaque partie du réseau de Petri temporel est décrite en langage SystemVerilog comme montré sur la Figure III.4. Elles sont connectées avec les signaux de communications intégrés dans une interface SystemVerilog. Cette interface générique permet à l'utilisateur de facilement connecter les différents blocs. De plus, si le concepteur souhaite changer le protocole de communication, le seul bloc à changer est celui de la transition, les connexions restant inchangées, comme montré sur la Figure III.3.

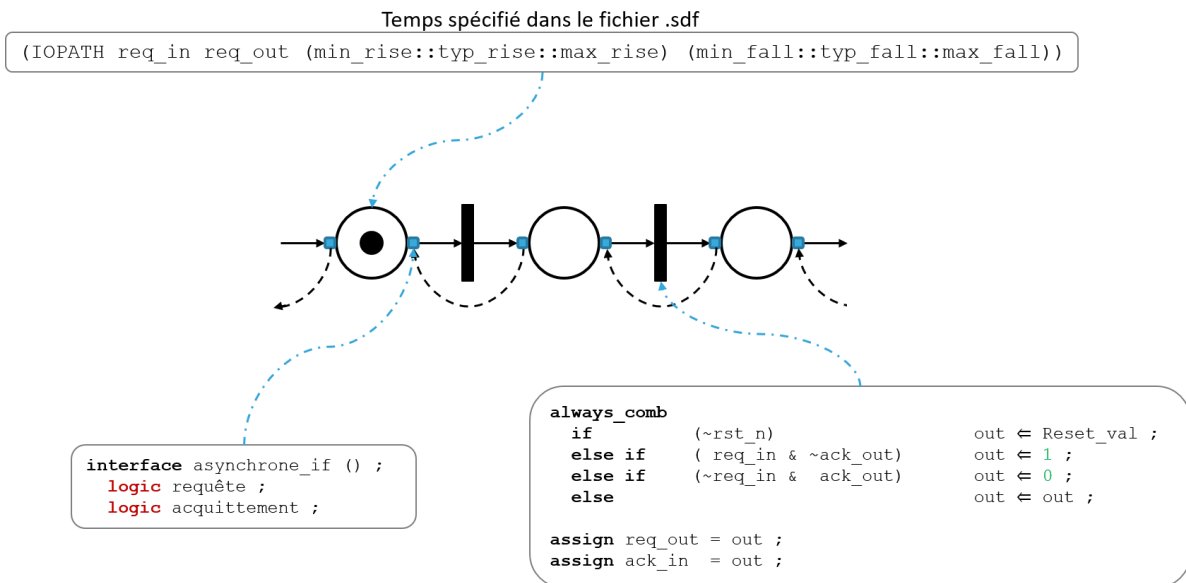


Figure III.4. Représentation du réseau de Petri Temporel avec un exemple de la spécification temporelle décrite dans le fichier *SDF*, de l'interface d'une place avec le canal de communication composé des signaux de requête et d'acquiescement et de la description d'une transition pour un protocole 4 phases.

Une fois le circuit de contrôle modélisé en SystemVerilog, il est possible de le simuler pour obtenir des informations sur son fonctionnement. La description en SystemVerilog nous permet d'utiliser les outils standards de conception synchrone. Un banc de test envoyant une requête sur l'entrée du circuit fournit le délai réellement écoulé avant que les transitions s'activent. Ce banc de test contient aussi un moniteur qui extrait les instants de commutations des registres et les mémorise dans un fichier.

Enfin, un pic de courant est associé à chaque instant de commutation pour obtenir la courbe de consommation de courant du circuit. Notre méthode de modélisation des pics de courant est décrite dans la partie suivante.

III. 3. Estimation du courant

Dans un circuit synchrone, la consommation de courant peut être modélisée en s'appuyant sur trois considérations :

- Dans un circuit CMOS, la commutation d'une porte produit une consommation de courant.
- La plupart de l'activité liée à la commutation des portes est localisée juste après les fronts du signal d'horloge.
- Les commutations des bascules et de l'arbre d'horloge produisent une part importante de la consommation totale de courant.

C'est pourquoi, les pics de courant sont uniformément distribués dans le temps, sur les fronts de l'horloge, et génèrent des harmoniques dans le spectre du circuit. La consommation de courant, d'un circuit numérique synchrone en logique CMOS peut être modélisée avec des

impulsions de courant au niveau des fronts du signal d'horloge, comme le montre la Figure III.5.

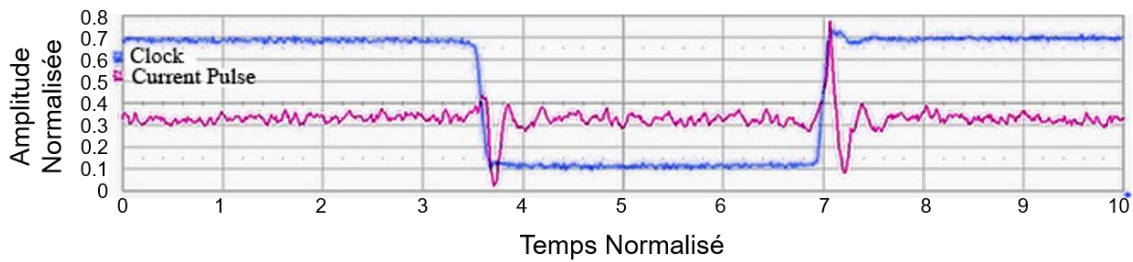


Figure III.5. Consommation de courant pour un circuit synchrone en CMOS avec les pics de courant sur les fronts du signal d'horloge.

Dans un circuit asynchrone, le signal global d'horloge est retiré. La commutation des bascules du circuit est déclenchée par des contrôleurs locaux asynchrones. Cependant, la forme de la consommation de courant d'une bascule reste la même et seule sa position dans le temps change. En conséquence, la distribution n'est plus uniforme ce qui permet de réduire le spectre.

La courbe de la consommation de courant dans le temps d'un circuit asynchrone peut être modélisée par des pics de courant aux instants de commutation des bascules.

III. 3. 1. Modèle d'un pic de courant

La plupart du temps, un modèle triangulaire est utilisé pour représenter un pic de courant [47], [48], [50]. Pour être plus précis dans notre méthode, nous avons décidé d'utiliser un modèle basé sur deux exponentielles dos à dos. Ce modèle est inspiré de la fonction de densité de probabilité de la distribution asymétrique de Laplace. En effet, cette fonction est composée de deux exponentielles dos à dos, elle est donnée dans l'Équation III-1, dans laquelle m est le paramètre de localisation de la fonction, k est le paramètre d'asymétrie et λ est le paramètre d'étalement. Le paramètre d'asymétrie de la fonction relie le temps de montée de la première exponentielle au temps de descente de la deuxième.

$$f(x) = \frac{\lambda}{k + 1/k} \begin{cases} e^{\frac{\lambda}{k}(x-m)} & \text{pour } x < m \\ e^{-\lambda k(x-m)} & \text{pour } x \geq m \end{cases} \quad \text{Équation III-1}$$

La Figure III.6 représente la fonction de densité de probabilité de la distribution asymétrique de Laplace, centrée en 0, pour trois paramètres d'asymétrie différents et pour un étalement λ de 1. Lorsque le paramètre d'asymétrie vaut 1, représenté par la courbe bleue, la fonction est symétrique et nous retrouvons la densité de probabilité de la distribution de Laplace. Les courbes verte et rouge sont asymétriques ($k \neq 1$) et elles sont un miroir l'une de l'autre.

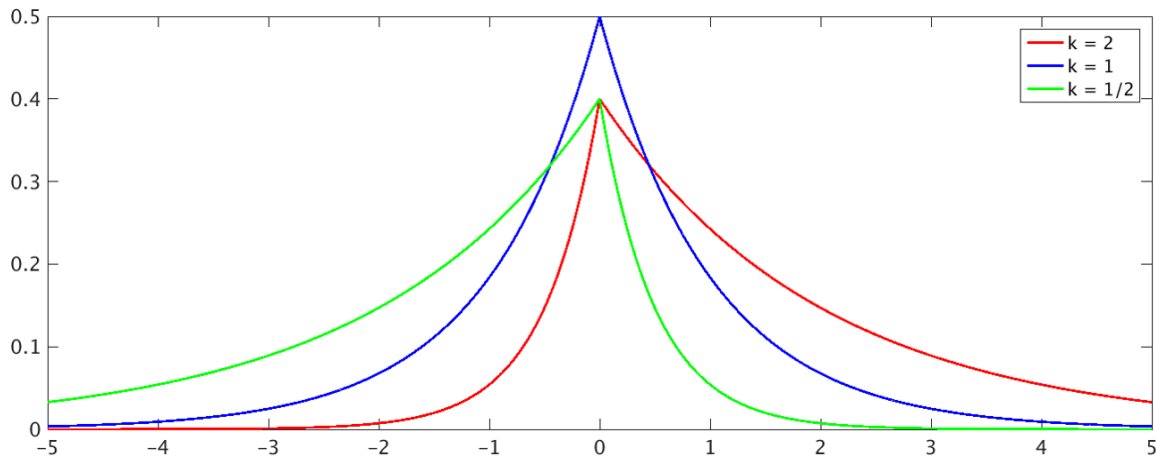


Figure III.6. Fonction de densité de probabilité de la distribution asymétrique de Laplace, centrée en 0, pour trois paramètres d'asymétrie différents et un étalement λ de 1.

Dans notre cas, le temps de montée de la première exponentielle n'est pas lié au temps de descente de la deuxième. Elles ont chacune leur propre coefficient. Notons, b , le coefficient de croissance de la première exponentielle et c le coefficient de décroissance de la deuxième. L'Équation III-2 exprime la fonction du modèle de courant pour un pic, où A est l'amplitude du pic et t_{com} est l'instant de commutation, où la consommation est la plus forte pour un étage du circuit.

$$f(x) = A \begin{cases} e^{b(x-t_{com})} & \text{pour } x \leq t_{com} \\ e^{-c(x-t_{com})} & \text{pour } x \geq t_{com} \end{cases} \quad \text{Équation III-2}$$

Les constantes de temps des pics de courant du circuit sont définies pour chaque étage grâce à une extraction de délai pendant l'étape d'analyse temporelle statique. Le temps de montée de la première exponentielle, qui compose notre pic de courant, est calculé à partir du délai du signal de requête depuis le contrôleur jusqu'aux registres qu'il contrôle. Ce délai est représenté par D_{prop_req} dans la Figure III.7. Il dépend du nombre d'éléments de mémorisation qui sont contrôlés par le contrôleur. En effet, plus ce nombre est important, plus l'arbre de propagation du signal de requête sera long afin d'assurer que les registres mémorisent en même temps les données d'un même étage.

Le nombre de registres détermine également la hauteur du pic de courant. La consommation maximale d'un étage du circuit se produit lorsque toutes les données de l'étage sont mémorisées. La consommation de la mémorisation d'un registre est déterminée grâce à la fiche technique de la librairie utilisée ou à l'aide d'une simulation électrique. Ensuite, cette valeur est multipliée par le nombre de registres dans l'étage, la valeur de $N_{registres}$ est obtenue. Ainsi, plus l'amplitude du pic est élevée, plus le temps de montée de la première exponentielle sera long. Cette estimation de l'amplitude permet donc de prendre en compte la consommation des registres et également celle de l'arbre de requête.

Enfin, la seconde exponentielle représente la consommation de courant de la logique combinatoire après les registres. Le temps de descente correspond au délai du chemin critique de l'étage, représenté par $D_{chemin_critique}$ dans la Figure III.7.

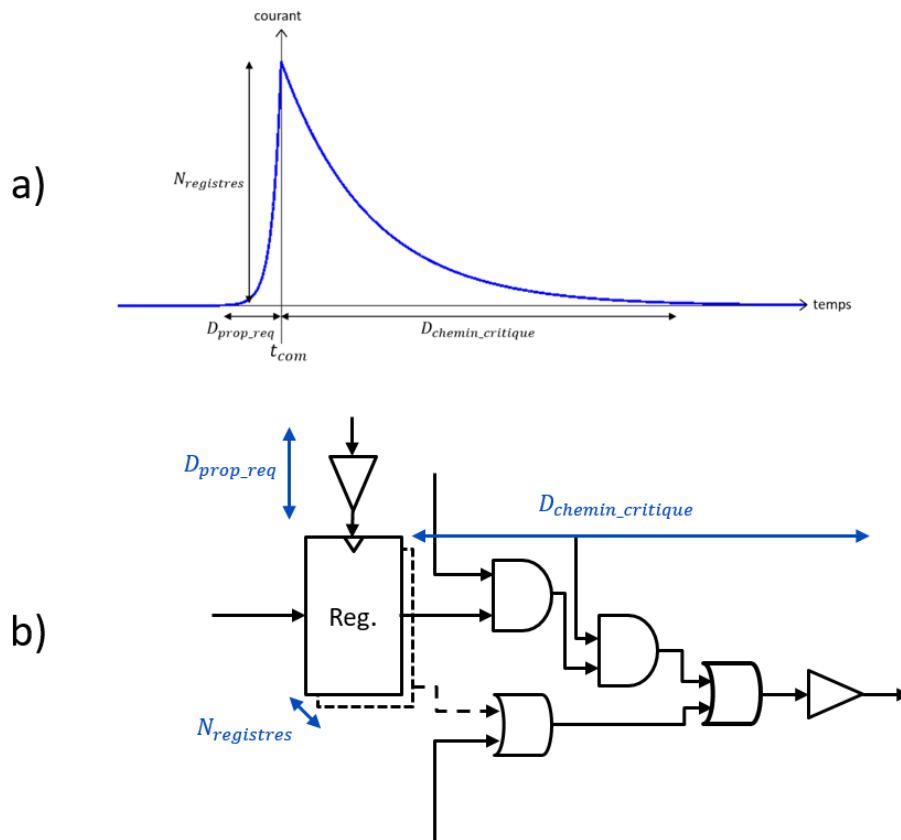


Figure III.7. Représentation a) d'un pic de courant avec le modèle de la double exponentielle asymétrique et b) d'un étage d'un circuit micropipeline permettant de trouver les paramètres du pic de courant.

A partir de ces informations de délais qui caractérisent le pic de courant, nous pouvons en déduire les coefficients à appliquer aux exponentielles pour obtenir le modèle du pic. Pour cela, nous définissons une amplitude minimale désirée aux points de début et de fin de la consommation de l'étage, respectivement « $-D_{prop_req}$ » et « $D_{chemin_critique}$ ». Ici, nous avons choisi 0,01 comme valeur minimale. De plus, l'amplitude du pic doit atteindre la valeur $N_{registres}$ à l'instant de commutation des registres, nous pouvons remplacer A par $N_{registres}$ dans l'Équation III-2.

Pour calculer le coefficient de la première exponentielle, x est remplacé par $(t_{com} - D_{prop_req})$ dans la partie de l'Équation III-2 où $x \leq t_{com}$, avec t_{com} le temps de commutation de l'étage. Nous souhaitons, alors, trouver le coefficient b qui résout l'Équation III-3.

$$N_{registres} * e^{b*(-D_{prop_req})} = 0,01 \quad \text{Équation III-3}$$

L'Équation III-4 nous donne la valeur du coefficient b de la première exponentielle.

$$b = - \frac{\log(0,01/N_{registers})}{D_{prop_req}} \quad \text{Équation III-4}$$

Pour la deuxième exponentielle, nous souhaitons trouver le coefficient c qui permettra d'avoir une amplitude de $0,01$ pour $x = D_{chemin_critique}$. Ainsi, c doit résoudre l'Équation III-5.

$$N_{registers} * e^{-c*D_{chemin_critique}} = 0,01 \quad \text{Équation III-5}$$

L'Équation III-6 nous donne la valeur du coefficient c .

$$c = - \frac{\log(0,01/N_{registers})}{D_{chemin_critique}} \quad \text{Équation III-6}$$

Les coefficients des exponentielles sont calculés pour chaque étage à partir des valeurs de D_{prop_req} , $N_{registers}$, et $D_{chemin_critique}$ extraites lors de l'analyse temporelle statique.

Les deux exponentielles de ce modèle permettent de prendre en compte à la fois la logique combinatoire et la mémorisation des données dans le circuit. Elles nous permettent d'avoir un modèle de haut niveau des pics en courant proche de la réalité. De plus, l'utilisation de ce modèle ne complexifie pas les calculs, pour obtenir la courbe en courant et le spectre électromagnétique, par rapport à l'utilisation d'un modèle triangulaire. On gagne donc en précision sans augmenter la complexité de calcul.

Pour montrer la pertinence de ce modèle, un étage de registres avec la propagation du signal de requête et une partie combinatoire très simple a été simulé avec l'outil Cadence Spectre®. La courbe bleue de la Figure III.8 représente la consommation de courant de l'étage après le front montant du signal de requête (la courbe rouge). La courbe verte représente le pic de courant modélisé avec un triangle. La courbe noire correspond à notre modèle du pic de courant de haut niveau, composé de deux exponentielles dos à dos. Nous constatons que la consommation de l'étage peut être assimilée à notre modèle et que le modèle triangulaire est plus pessimiste.

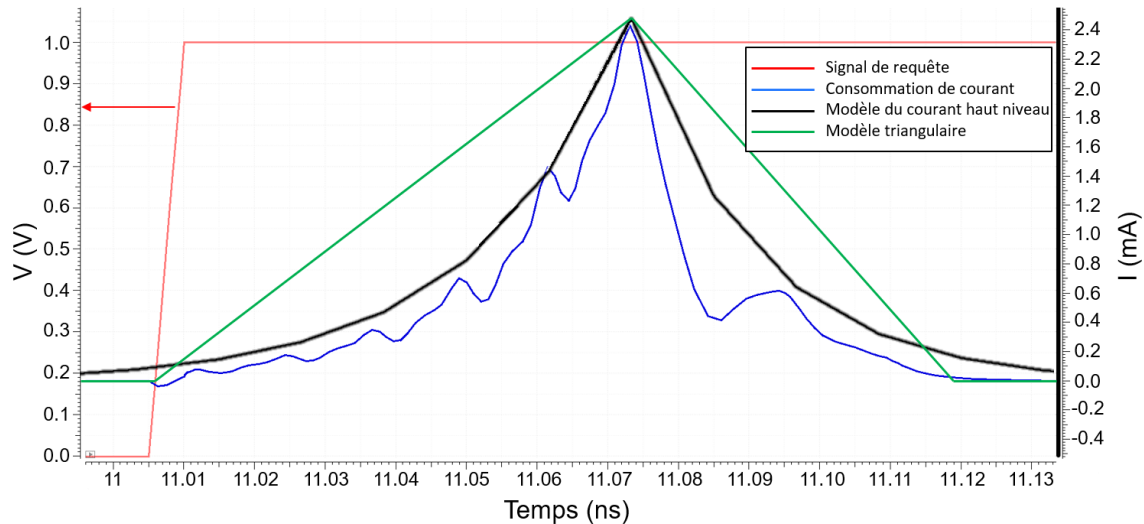


Figure III.8. Simulation Spectre® d'un étage de registres avec la propagation du signal de requête et une partie combinatoire simple. La courbe rouge représente le font montant de la requête, la bleue la consommation de courant de l'étage, la noire notre modèle de haut niveau d'un pic de courant et la verte le modèle triangulaire.

Notre modèle nous permet de déterminer la forme des pics de courant pour chaque étage du circuit. En effet, chaque étage effectue un traitement différent sur les données et peut contenir un nombre de registres différents. Toutes les valeurs des délais sont extraites durant l'analyse temporelle statique du circuit. Ainsi, nous obtenons un modèle de pic de courant pour chaque étage pour ensuite pouvoir modéliser la consommation de courant totale du circuit.

III. 3. 2. Modèle de la consommation de courant

Le modèle de la courbe de consommation de courant totale est obtenu en associant le résultat du réseau de Petri temporel avec l'estimation des pics de courant pour chaque étage du circuit micropipeline. En effet, à chaque instant de commutation, déterminé grâce à la simulation du réseau de Petri temporel, est associé le pic de courant correspondant, déterminé grâce à l'analyse temporelle du circuit. Ainsi, nous obtenons le modèle de la consommation de courant du circuit. Cette méthode nous permet d'avoir un modèle rapide et proche de la réalité pour ensuite estimer le rayonnement électromagnétique du circuit.

La Figure III.9 représente la consommation de courant totale pour la propagation d'un jeton dans un circuit micropipeline de quatre étages. Les quatre instants de commutation ont été extraits suite à la simulation du réseau de Petri temporel associé au circuit. Ensuite, un pic de courant est associé à chaque instant. Chaque pic a ses paramètres liés aux propriétés de son étage dans le circuit, temps de propagation de la requête, nombre de registres et délai du chemin critique.

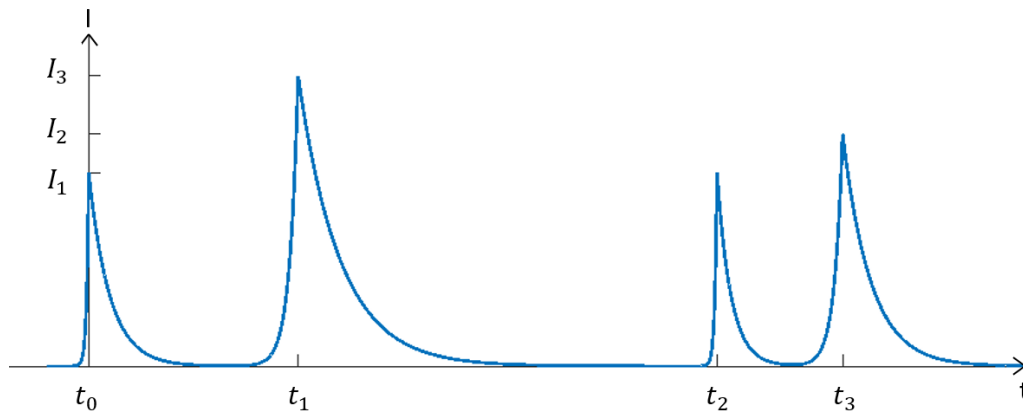


Figure III.9. Modèle de consommation de courant pour un circuit micropipeline de quatre étages.

III. 4. Conclusion

Dans la première partie de ce chapitre, les réseaux de Petri ont été présentés. Ils permettent d'avoir une représentation graphique du réseau de contrôle du circuit micropipeline. Les canaux de communication et les contrôleurs sont représentés par des barres verticales, appelées transitions. La logique combinatoire et le délai qu'elle représente entre deux contrôleurs sont représentés par un cercle, appelé place. Ainsi, le réseau de Petri est obligatoirement représenté par une alternance de transitions et de places. Des jetons dans les places permettent de représenter les ressources utilisées au cours de l'exécution du modèle. Ce sont les transitions qui permettent de faire avancer les ressources d'une place à l'autre. Il est possible d'ajouter une notion temporelle aux réseaux de Petri. Dans notre cas, une valeur de délai à respecter avant le passage d'une donnée, d'une place à l'autre, est spécifiée pour chaque place. Ainsi, pour qu'une transition s'active trois règles doivent être respectées. Premièrement, toutes les places en entrée de la transition doivent avoir un jeton. Deuxièmement, le délai de chaque place doit être écoulé. Enfin, toutes les places en sortie doivent être libres. Nous avons décidé d'implémenter notre réseau de Petri temporel en SystemVerilog afin de profiter d'un simulateur standard basé sur des événements. Ainsi, grâce au fichier de test, décrit également en SystemVerilog, nous pouvons extraire les temps d'activations du réseau de Petri temporel. Ces valeurs correspondent aux temps de commutation des éléments de mémorisation des différents étages du circuit micropipeline. Ils vont ensuite nous permettre d'obtenir la consommation totale de courant du circuit.

Dans la deuxième partie, nous avons vu la modélisation de la consommation de courant à partir des temps de commutation des étages du circuit micropipeline. Un nouveau modèle pour les pics de courant des étages a été établi. Un modèle triangulaire est généralement utilisé pour représenter le courant consommé lors de la mémorisation et du traitement des données d'un étage. Pour être plus proche de la réalité, nous proposons d'utiliser un modèle composé de deux exponentielles dos à dos. La première permet de prendre en compte le

temps de la propagation du signal de requête depuis le contrôleur vers les éléments de mémorisation. Le point le plus haut du pic dépend du nombre de bascules dans l'étage. La consommation d'un étage arrive à son maximum lorsque toutes les bascules commutent pour mémoriser les données. Enfin, la deuxième exponentielle correspond à la consommation de courant de la logique combinatoire après les registres. Le temps de descente dépend du chemin critique de la logique associée à l'étage. Ainsi, à partir de l'analyse temporelle statique du circuit, nous pouvons extraire les paramètres des pics de courant de chaque étage. La consommation de courant totale du circuit micropipeline est obtenue en associant, à chaque instant de commutation de l'étage, le pic de courant correspondant. Cette association nous permet d'obtenir un simulateur analogique rapide de la consommation de courant de notre circuit.

La modélisation de haut niveau du courant nous permet d'avoir une estimation rapide de la consommation et ainsi d'en déduire le spectre électromagnétique du circuit. Nous avons vu dans le chapitre précédent que cette transformation, temporelle vers fréquentielle est effectuée grâce à une transformée de Fourier. Cela permet ensuite à l'algorithme génétique de déterminer les délais du circuit pour que le spectre respecte un gabarit. Ainsi, il est nécessaire d'avoir une communication entre l'algorithme génétique, qui peut ajouter du délai entre les contrôleurs pour modifier le spectre, et le simulateur analogique rapide du courant qui permet d'obtenir la consommation de courant, nécessaire à l'algorithme pour calculer le spectre électromagnétique. L'interaction entre ces deux étapes sera décrite dans le chapitre suivant à travers la présentation de notre flot de conception complet pour le contrôle des émissions électromagnétiques.

Chapitre IV.

Flot de conception

IV. 1.	Introduction.....	64
IV. 2.	Présentation du flot de conception.....	65
IV. 3.	Conception du circuit	66
IV. 4.	Synthèse logique.....	68
IV. 4. 1.	Etude du réseau de contrôle.....	68
IV. 4. 2.	Rupture des boucles combinatoires	70
IV. 4. 3.	Définition des horloges.....	72
IV. 5.	Analyse temporelle statique.....	74
IV. 5. 1.	Analyse des chemins	76
IV. 5. 2.	Récupération des délais.....	77
IV. 6.	Co-Simulation	79
IV. 6. 1.	Algorithme Génétique	79
IV. 6. 2.	Simulation Analogique Rapide.....	80
IV. 6. 3.	Obtention d'une solution.....	81
IV. 7.	Implémentation	83
IV. 7. 1.	Implémentation synchrone.....	83
IV. 7. 2.	Implémentation asynchrone.....	83
IV. 8.	Conclusion	84

Résumé :

Nous avons défini un flot de conception, qui utilise les outils de Conception Assistée par Ordinateur (CAO) standards, pour contrôler le rayonnement électromagnétique des circuits numériques avec un gabarit défini par le concepteur. Le circuit est décrit dans un langage de description de matériel au niveau registres (*Register Transfert Level, RTL*) en utilisant la méthode de conception des circuits micropipelines. Le gabarit fréquentiel est défini lors de l'étape de conception.

La synthèse logique permet de traduire la description RTL en portes logiques de la librairie. Ensuite, l'analyse temporelle statique permet de récupérer les valeurs de délais du circuit. L'étape de co-simulation permet de trouver les valeurs de délais du chemin de requête pour que le spectre respecte le gabarit. Pour cela, l'algorithme génétique crée des combinaisons de délais. Celles-ci sont utilisées pour simuler le circuit et récupérer les instants de commutation des étages. A ces instants sont associés les pics de courant des étages pour obtenir la consommation du circuit. L'algorithme génétique compare le spectre de chaque combinaison avec le gabarit, jusqu'à obtenir une solution. Enfin, le circuit est implémenté, en gardant des outils de CAO standards, avec les valeurs finales des délais.

IV. 1. Introduction

Ce chapitre a pour but de présenter notre flot de conception qui permet de contrôler le rayonnement électromagnétique des circuits numériques. Un exemple de circuit sera utilisé tout au long de ce chapitre. Il permettra d'illustrer le déroulement de notre flot de conception.

La première partie présente l'enchaînement des différentes étapes à travers un schéma du flot de conception. Les parties suivantes présentent ces étapes de façon détaillée.

Tout d'abord, la description du circuit, qui utilise le SystemVerilog comme langage de description matérielle, est présentée. Ce langage standard, est utilisé pour décrire et vérifier des circuits. C'est également dans cette étape que le concepteur définit le gabarit fréquentiel que le circuit doit respecter. Ce dernier est utilisé dans l'étape de mise en forme du spectre.

La partie suivante présente la synthèse logique du circuit. Cette étape permet de traduire la description SystemVerilog du circuit en portes logiques de la librairie spécifiée. Pour cela, le concepteur spécifie des contraintes qui permettent de piloter l'outil dans ses phases d'optimisation. Dans notre flot de conception, une étape initiale est ajoutée par rapport à une synthèse logique classique pour analyser le réseau de contrôle du circuit et déterminer les éléments de contrôle présents. Cette étude nous permet de couper les boucles combinatoires qui ne sont pas compatibles avec la synthèse logique et de définir les horloges du circuit. Une fois ces contraintes appliquées au circuit, la conversion peut être réalisée.

Ensuite, l'analyse temporelle statique du circuit est présentée. Cette étape utilise le circuit décrit au niveau des portes physiques ainsi que les contraintes utilisées lors de la synthèse. Ces contraintes sont enrichies pour pouvoir analyser tous les chemins du circuit. C'est à partir de ces contraintes que les valeurs minimales des délais du chemin de contrôle sont extraites.

Une fois ces valeurs extraites, nous pouvons passer à l'étape de mise en forme du spectre électromagnétique. La co-simulation, entre l'algorithme génétique et la simulation analogique rapide, permet de trouver les délais qui assurent le respect du gabarit. L'algorithme génétique génère une combinaison de délais pour le circuit. Celle-ci est ensuite simulée avec notre outil de simulation analogique rapide pour obtenir la consommation de courant du circuit. Cette courbe est envoyée à l'algorithme génétique qui, grâce à la transformée de Fourier du courant, évalue le spectre électromagnétique en le comparant en tout point avec le gabarit. Si une solution est trouvée le processus s'arrête. Sinon, l'algorithme génère une nouvelle combinaison de délais.

Enfin, une fois une solution trouvée, le circuit peut être placé et routé avec cette combinaison de délais. Tout d'abord, la surface du circuit est délimitée avec les zones des différentes parties. Ensuite, les cellules du circuit, en dehors de celles de l'arbre du signal de requête, sont placées. L'outil détermine le nombre de cellules nécessaires pour équilibrer les chemins entre les contrôleurs et les éléments de mémorisation et il les place. Il ajoute par la suite les délais entre les contrôleurs pour obtenir les valeurs déterminées lors de la co-simulation. Enfin, le signal de requête puis les autres connexions du circuit sont routés. Des

lignes de délais programmables peuvent être ajoutées sur le chemin de la requête pour réajuster, après la fabrication du circuit, les délais aux valeurs désirées. Cela permet de prendre en compte les variations des procédés de fabrication.

IV. 2. Présentation du flot de conception

Un flot de conception a été développé de manière à contrôler le rayonnement électromagnétique des circuits numériques avec un simulateur analogique dédié qui permet de modéliser la consommation totale de courant du circuit. La Figure IV.1 présente le déroulement du flot de conception.

Toutes les étapes du flot sont réalisées avec des outils commerciaux standards pour bénéficier de la puissance des outils de conception actuels. De plus, cela permet de faciliter son utilisation pour des concepteurs synchrones et d'éviter le développement d'outils spécifiques, qui nécessiteraient un temps de développement, d'adaptation et de prise en main.

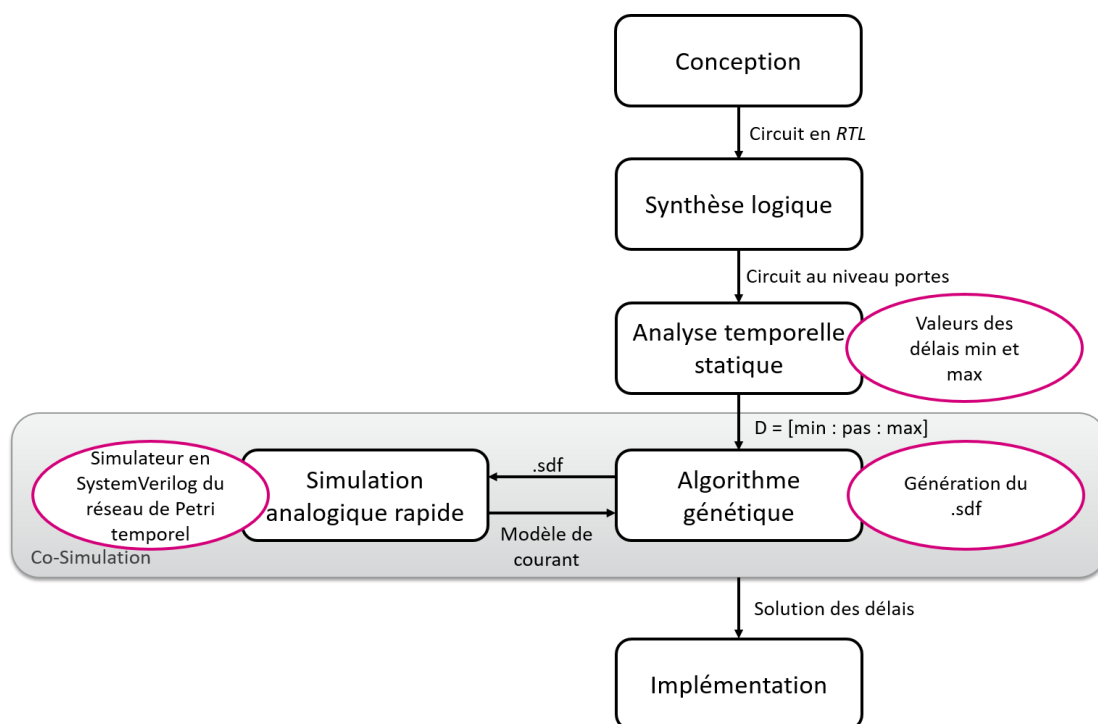


Figure IV.1. Flot de conception pour contrôler le rayonnement électromagnétique d'un circuit numérique.

La première étape consiste à décrire le circuit, avec son chemin de données et de contrôle, en utilisant un langage de description d'architecture microélectronique au niveau registres (*Register Transfer Level, RTL*). A partir de cette description la synthèse logique est réalisée pour obtenir la description au niveau portes logiques du circuit. Celle-ci est utilisée pour réaliser l'analyse temporelle statique (*STA*) du circuit, afin de récupérer les différentes valeurs de délais nécessaires pour les étapes suivantes. Vient ensuite, l'étape de mise en forme du

rayonnement électromagnétique, représentée par le rectangle gris dans la Figure IV.1. L'algorithme génétique crée des combinaisons de délais pour le chemin de contrôle. Chaque combinaison est simulée grâce au réseau de Petri temporel pour obtenir les temps de commutation de tous les étages. Les pics de courant des étages sont ensuite associés à ces temps pour obtenir la consommation de courant du circuit. Cette courbe est envoyée à l'algorithme génétique. La transformée de Fourier est calculée pour obtenir le spectre du circuit. Celui-ci est comparé au gabarit défini par le concepteur pour évaluer si les valeurs des délais répondent au problème. Si ce n'est pas le cas, une nouvelle itération est effectuée jusqu'à l'obtention d'une combinaison de délais qui permet d'avoir un rayonnement en dessous du gabarit. Enfin, la dernière étape consiste à réaliser l'implémentation physique du circuit.

Les différentes étapes de flot de conception, représentées dans la Figure IV.1, sont décrites en détails dans les parties suivantes. Elles seront illustrées à l'aide d'un circuit d'exemple développé avec notre flot.

IV. 3. Conception du circuit

La première étape du flot est naturellement de concevoir le circuit numérique en logique asynchrone, ici en micropipeline. Cette classe de circuit est utilisée car son architecture est très proche des circuits synchrones, donc plus facile à manipuler pour un concepteur avec une formation synchrone. Pour concevoir un circuit micropipeline, il faut décrire le chemin de données où tous les traitements sont effectués, comme dans un circuit synchrone. Il faut également concevoir le chemin de contrôle qui permet de cadencer l'avancement des données pour les calculs. Celui-ci remplace le signal d'horloge. Une bibliothèque d'éléments de contrôle a été élaborée. Elle est composée d'un contrôleur pour un protocole quatre phases mais également d'éléments nécessaires pour réaliser un circuit non-linéaire. Cette bibliothèque pourra être enrichie par la suite pour supporter les autres protocoles ou classes de circuits asynchrones.

Nous avons vu dans le chapitre II qu'il est nécessaire d'ajouter un délai sur le signal de requête pour prendre en compte le temps de la logique combinatoire. Néanmoins, ce n'est pas au cours de cette première étape du flot de conception que ces délais sont insérés. La valeur minimale à ajouter pour capturer une donnée valide dans les registres n'est pas encore connue. Surtout nous ne connaissons pas la valeur qui permet de contrôler le rayonnement électromagnétique du circuit. Nous allons seulement ajouter une cellule tampon, appelée *buffer*, pour représenter et localiser l'emplacement du futur délai.

Au cours de ce chapitre nous allons illustrer le flot de conception à l'aide du circuit représenté sur la Figure IV.2. La partie contrôle du circuit est représentée en haut de la figure tandis que le chemin de données est en bas. Nous constatons que chaque étage de registres est cadencé par un contrôleur. Le circuit que nous avons choisi est non-linéaire pour illustrer

l'applicabilité de notre méthode à tous les types de circuits. La porte appelée « fourche » (*fork* en anglais) permet d'envoyer la requête aux deux contrôleurs et surtout d'attendre l'acquittement des deux chemins avant de l'envoyer au premier contrôleur. La porte « jonction » (*join* en anglais) permet d'attendre les deux requêtes avant de les envoyer au dernier contrôleur et d'envoyer l'acquittement de celui-ci directement aux deux chemins. Ces deux portes sont conçues grâce à des portes de Muller. Le premier et le dernier étage possèdent seize registres chacun tandis que les deux étages concurrents, entre la fourche et la jonction, en possèdent huit chacun. Ces différences permettront d'illustrer la différence de hauteur des pics de courant en fonction du nombre d'éléments de mémorisation.

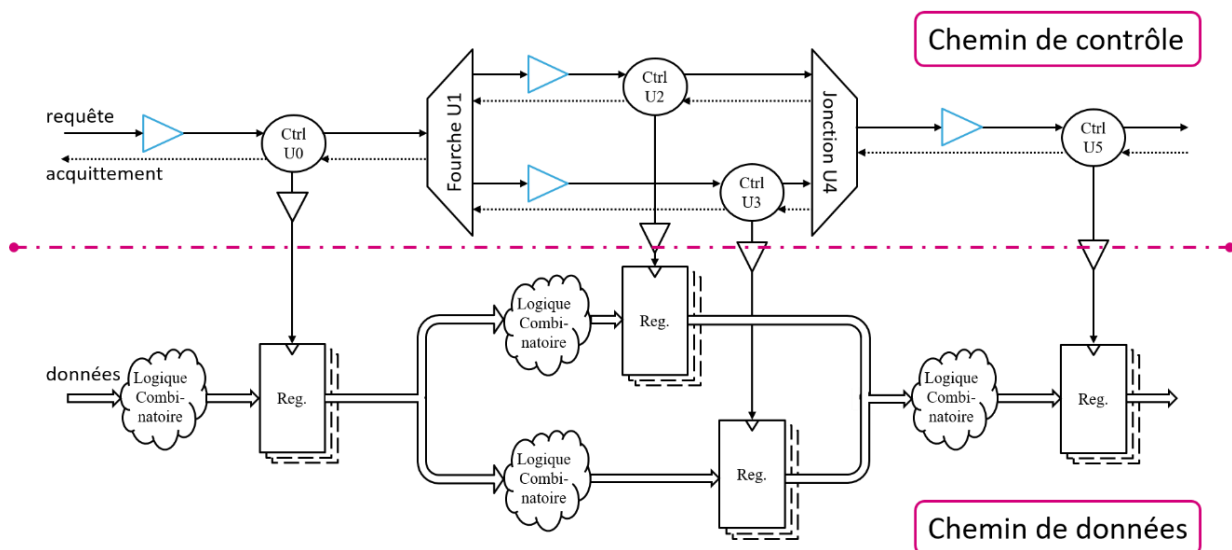


Figure IV.2. Exemple de circuit micropipeline, utilisé pour illustrer ce chapitre.

C'est également au cours de cette étape que le concepteur va définir le gabarit en fréquences que le circuit doit respecter. Celui-ci peut être extrait d'une norme existante ou il peut être créé par le concepteur. Comme nous n'avons pas de norme à respecter dans notre exemple, nous avons décidé de simuler notre circuit avec les mêmes délais entre chaque contrôleur. Cette configuration correspond au pire cas pour un circuit asynchrone. Les appels de courant étant répartis périodiquement, les harmoniques sont à nouveau présentes dans le spectre. Ensuite, nous avons décidé d'optimiser en particulier les basses fréquences. La Figure IV.3 montre le spectre électromagnétique pour des délais de 1,6 ns entre chaque contrôleur, nous verrons dans l'étape d'analyse temporelle statique que ce délai est suffisant pour que le circuit fonctionne correctement. Nous avons ensuite défini le gabarit représenté en rouge sur la Figure IV.3. Il sera utilisé dans l'étape de mise en forme du spectre électromagnétique.

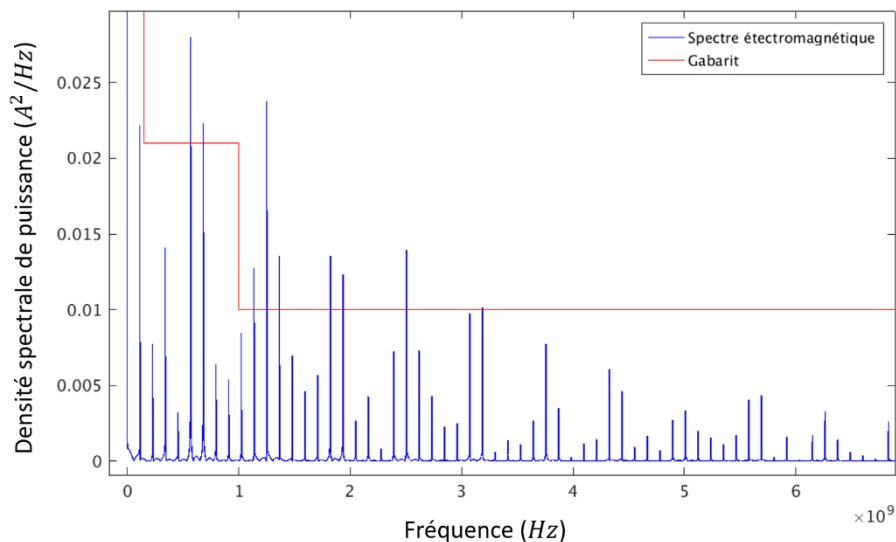


Figure IV.3. Spectre électromagnétique du circuit d'exemple, en bleu, avec des délais de 1,6 ns entre chaque contrôleur et définition du gabarit à respecter pour le circuit, courbe rouge.

Maintenant que le circuit est conçu et que le gabarit est défini, il faut déterminer les délais à insérer dans le chemin de contrôle pour que le rayonnement électromagnétique du circuit respecte le gabarit. Pour cela, le circuit décrit au niveau registres doit être traduit en portes logiques pour avoir les valeurs des délais dans les portes.

IV. 4. Synthèse logique

Cette étape du flot de conception permet de traduire la description au niveau registres du circuit en portes logiques de la librairie spécifiée par le concepteur. Cette étape est pilotée par des contraintes temporelles définies par le concepteur. Pour effectuer la synthèse en respectant ces contraintes, l'outil utilise un moteur d'analyse temporelle statique (*Static Timing Analysis, STA*). Cela lui permet de calculer les délais du circuit et d'optimiser le chemin de données. Le concepteur spécifie ses contraintes dans un script décrit en langage *Tcl* qui est ensuite chargé dans l'outil de synthèse. Le circuit ne doit pas posséder de boucles combinatoires pour pouvoir analyser les performances. Nous avons donc ajouté une étape d'analyse de réseau de contrôle afin d'automatiser la rupture des boucles combinatoires. Elle permet aussi la définition des horloges du système.

Les sous-parties qui suivent expliquent les étapes de la synthèse logique de notre flot. Elles sont toutes réalisées à l'aide de l'outil standard de synthèse, Design Compiler®, de l'entreprise Synopsys.

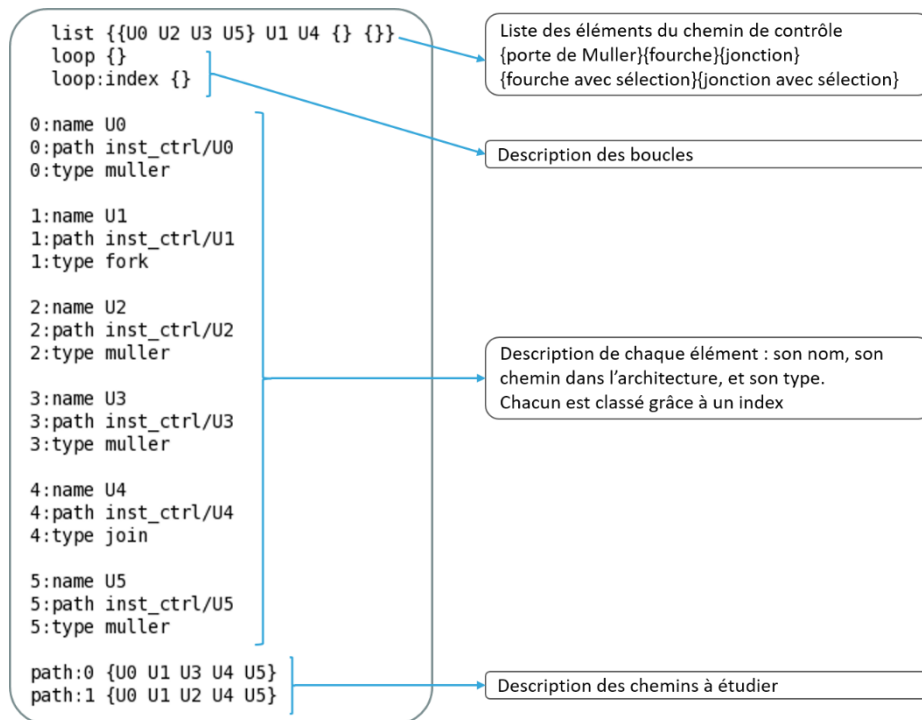
IV. 4. 1. Etude du réseau de contrôle

Une première étape avant la synthèse est de réaliser une analyse du réseau de contrôle du circuit. Cette étude du réseau a deux objectifs. Premièrement, elle permet de renvoyer une

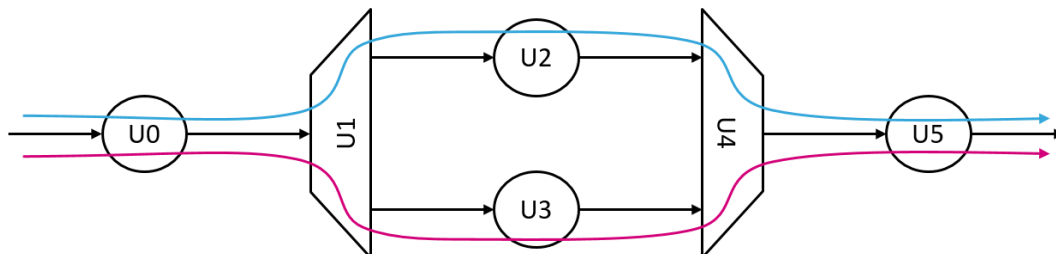
image du chemin de contrôle du circuit. Celle-ci est utilisée par le concepteur pour vérifier de façon visuelle que la description de son architecture correspond bien à sa spécification. Il peut par exemple s'apercevoir d'une mauvaise connexion du signal de requête et ainsi la corriger suffisamment tôt dans le flot de conception. Deuxièmement, elle permet de lister tous les contrôleurs du réseau ainsi que tous les chemins à analyser. Ces informations seront très utiles pour réaliser les étapes suivantes de la synthèse ainsi que pour l'analyse temporelle statique.

L'objectif de notre analyseur est de parcourir le chemin de contrôle et de répertorier les informations de celui-ci dans un tableau. Pour cela, nous allons suivre le chemin de la requête à travers le réseau de contrôle, depuis son port d'entrée jusqu'à son port de sortie. Chaque élément de contrôle rencontré est répertorié avec un index pour pouvoir récupérer ses informations facilement. Les différents chemins empruntés par le signal de requête sont également mémorisés pour les analyser dans l'étape d'analyse temporelle statique.

L'analyse de réseau a été appliquée au circuit décrit dans la partie précédente (Figure IV.2). La Figure IV.4 montre les informations extraites au cours de cette étape. Tout d'abord, une liste de tous les éléments de contrôle présents dans le circuit est décrite. Nous retrouvons les contrôleurs U0, U2, U3, U5, la fourche U1 et la jonction U4. Deux emplacements restent vides, ils servent à lister les fourches et les jonctions qui possèdent un signal de sélection pour choisir une branche parmi celles disponibles. Notre circuit ne possède pas ce type de porte dans son réseau de contrôle, la fourche et la jonction n'ont pas d'entrée de sélection et les signaux sont propagés dans les deux branches. La section suivante permet de connaître les informations sur les boucles dans le réseau de contrôle. Le circuit d'exemple n'en possède pas mais nous verrons un autre circuit, avec une boucle sur la requête, dans le chapitre suivant. Les deux éléments de contrôle où la boucle commence sont répertoriés dans la première ligne et leurs indices respectifs dans la deuxième. La partie suivante liste tous les éléments de contrôle du circuit. Pour chacun, leur nom, leur chemin dans la hiérarchie du circuit ainsi que leur type sont listés. Enfin, la dernière partie répertorie les chemins du réseau de contrôle, qui seront à étudier dans les prochaines étapes.



La Figure IV.5 montre un exemple de ce que pourrait être la sortie visuelle de notre outil d'analyse de réseau appliqué au circuit d'exemple utilisé dans ce chapitre (Figure IV.2). Les deux flèches, bleue et rose, matérialisent les deux chemins à étudier pour ce circuit, qui sont retournés sous forme de listes par l'outil, comme montré dans la dernière section de la Figure IV.4.



Grâce à l'analyse de réseau, toutes les informations pour couper les boucles combinatoires sont disponibles, sans que le concepteur ait besoin de décrire son chemin de contrôle dans un langage de script. Elles seront également utiles lors de l'étape de l'analyse temporelle statique.

IV. 4. 2. Rupture des boucles combinatoires

Pour pouvoir réaliser la synthèse logique, le circuit ne doit pas posséder de boucles combinatoires. Celles-ci ne peuvent pas être analysées de manière statique lorsque la synthèse évalue les performances. Or, dans les circuits asynchrones il y a des boucles

combinatoires entre les signaux de requêtes et d'acquittements. Dans notre cas, nous souhaitons étudier la propagation de la requête pour déterminer les délais à insérer entre les contrôleurs. Pour cela, les arcs du signal d'acquittement sont coupés dans tous les éléments de contrôle. Si la librairie utilisée ne contient pas de porte de Muller intégrée, il est nécessaire de concevoir la porte de Muller avec des portes standards. Dans ce cas, il sera nécessaire de casser les arcs internes qui créent une boucle combinatoire. Par exemple, à l'intérieur d'une porte de Muller réalisée avec des portes ET et OU, la sortie de la porte OU est rebouclée vers les entrées de deux portes ET, comme le montre la Figure IV.6.a. Il faut donc couper ces deux arcs. La Figure IV.6 montre les arcs qui doivent être coupés dans les contrôleurs, les fourches et les jonctions. Ils sont représentés en pointillés avec une croix rouge. Les arcs en bleus montrent la propagation du signal de requête dans les éléments. Dans cet exemple, nous utilisons le cas le plus complexe, c'est-à-dire sans porte de Muller intégrée.

L'analyse du réseau, effectuée précédemment, permet de connaître quels sont les éléments de contrôle présents ainsi que leur chemin hiérarchique dans le circuit. Avec ces informations, un script en *Tcl* a été développé pour automatiquement couper les boucles des différentes portes.

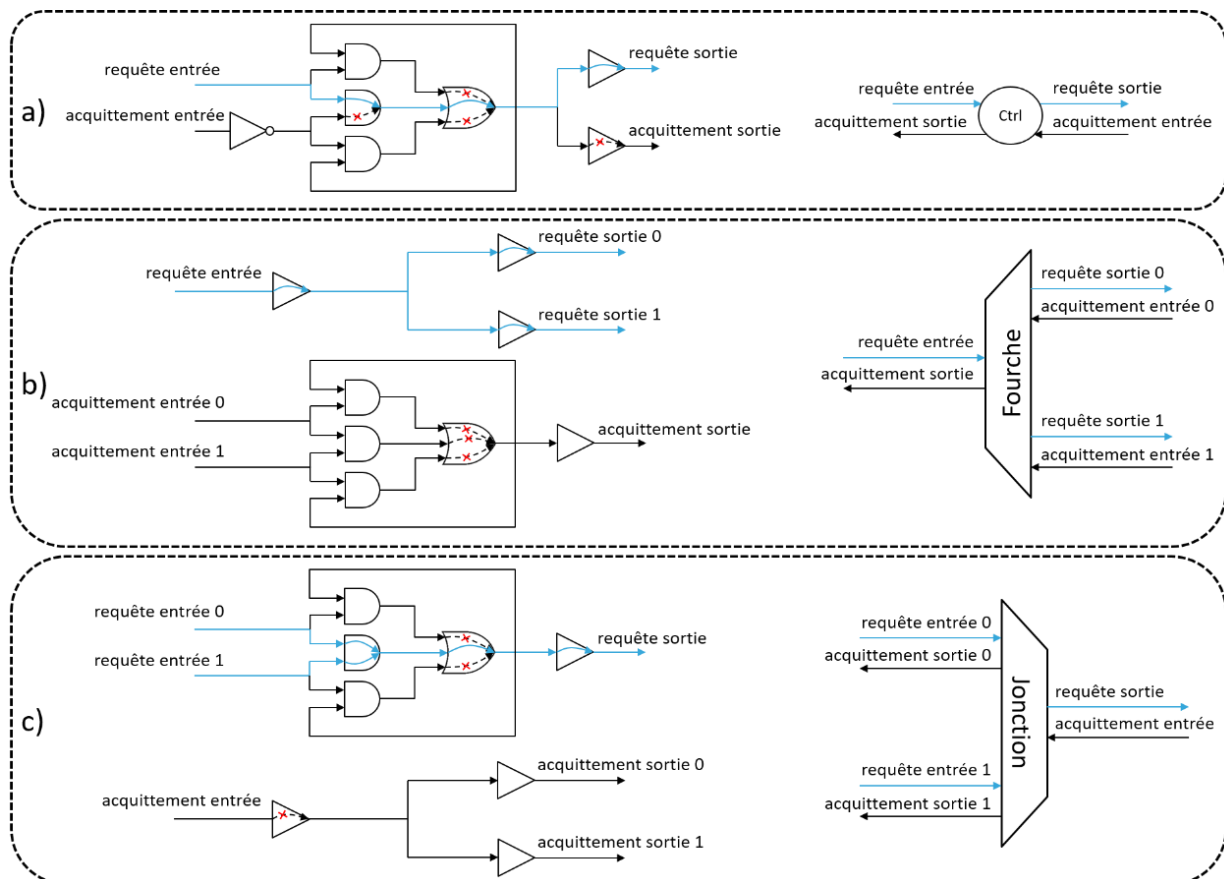


Figure IV.6. Représentation des arcs coupés dans le cas d'un contrôleur, d'une fourche et d'une jonction.

De plus, le chemin de contrôle peut posséder des boucles sur le signal de requête. Cette information est connue grâce à l'analyse de réseau. Ce cas peut être présent dans le réseau

de contrôle si, par exemple, le chemin de la requête se dédouble et se recombine, ce qui équivaut à avoir une fourche et une jonction dans le circuit, comme le montre la Figure IV.7. Dans cette configuration, l'analyse du réseau retourne l'information sur la fourche qui est concernée. Pour la synthèse, la boucle de retour est coupée dans la fourche, afin de garder le chemin principal. Dans la Figure IV.7, l'arc du buffer en bleu foncé est coupé.

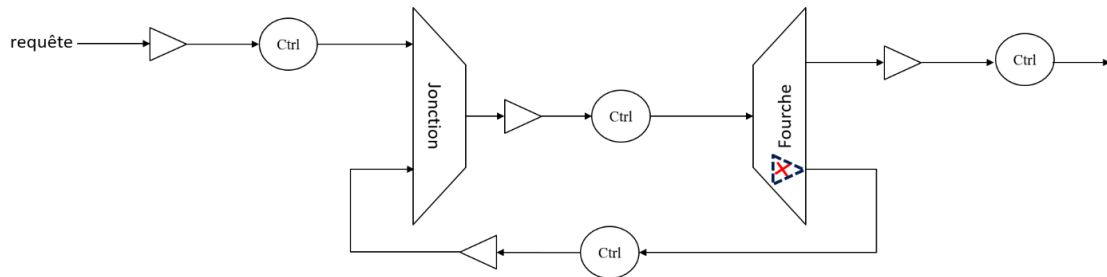


Figure IV.7. Représentation de la rupture de la boucle du signal de requête.

Après cette étape, toutes les boucles combinatoires du circuit sont coupées. Il contient maintenant seulement des chemins directs sur lesquels nous pouvons appliquer une contrainte d'horloge.

IV. 4. 3. Définition des horloges

La définition des horloges dans le circuit permet de déterminer les chemins qui sont dépendants et pour lesquels il faut respecter des valeurs temporelles pour ne pas avoir de violations. Cette étape est illustrée à l'aide de la Figure IV.8, dans le cas d'un circuit synchrone et asynchrone.

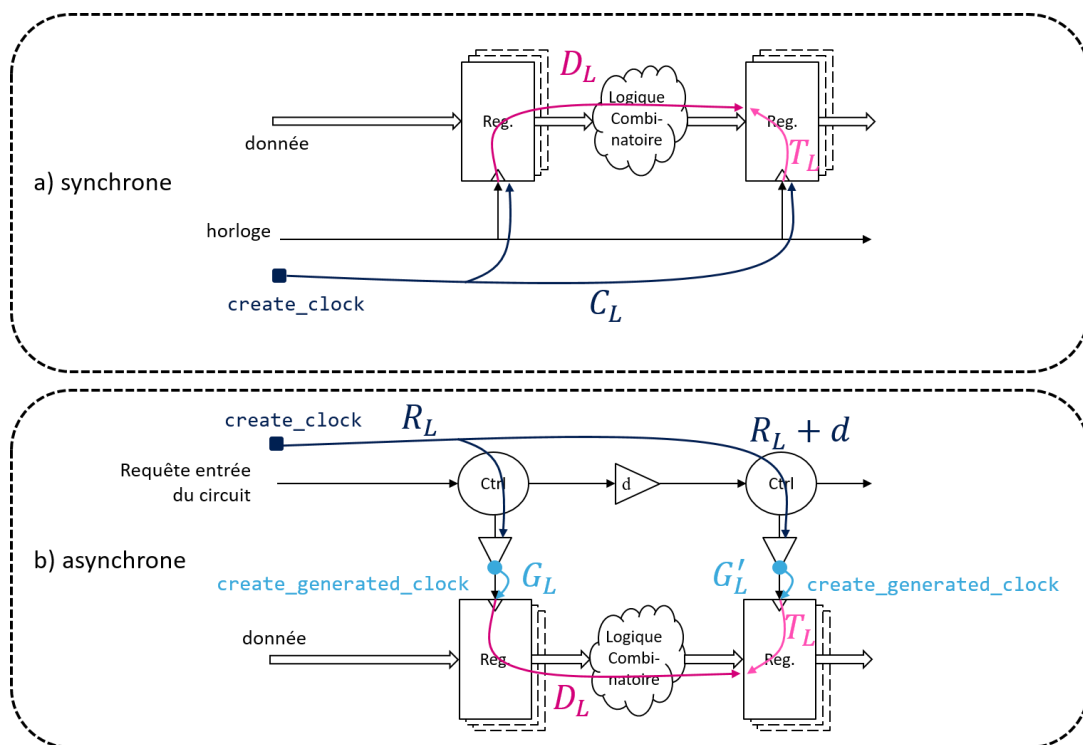


Figure IV.8. Définition des horloges dans un circuit a) synchrone et b) asynchrone.

Dans les circuits synchrones, une horloge globale est définie sur le port d'entrée de l'horloge à l'aide de la commande « `create_clock -period Tp` », comme le montre la Figure IV.8.a. Le concepteur peut ainsi contraindre la période de fonctionnement, T_p . Ensuite, l'horloge est propagée dans tout le circuit pour contrôler les registres. Lorsque l'outil va effectuer la synthèse logique, il va corriger les chemins de données qui ont une violation temporelle. Considérons l'étude du temps d'établissement de la donnée pour montrer le fonctionnement de l'optimisation de la synthèse. La donnée doit être stable avant une période d'horloge moins le temps d'établissement du registre. Nous obtenons l'Équation IV-1, où C_L correspond au temps de propagation de l'horloge jusqu'aux registres, T_p est la période de l'horloge, T_L le temps d'établissement du registre et D_L le temps maximum de propagation de la donnée dans la logique combinatoire.

$$C_L + T_p - T_L > C_L + D_L \quad \text{Équation IV-1}$$

Considérons que les chemins de propagation de l'horloge sont équilibrés pour tous les registres. En simplifiant C_L , l'Équation IV-2 nous donne la valeur maximale que les chemins de données doivent respecter pour que le circuit fonctionne correctement. C'est à partir de cette valeur que l'outil de synthèse va optimiser les chemins de données pour les circuits synchrones.

$$D_L < T_p - T_L \quad \text{Équation IV-2}$$

Dans les circuits asynchrones, l'outil de synthèse logique va également chercher à optimiser les chemins de données. Pour cela, une horloge globale est créée sur le port d'entrée de la requête du circuit, en utilisant la même commande que pour les circuits synchrones, comme le montre la Figure IV.8.b. Ensuite, des horloges, générées à partir de l'horloge globale, sont créées pour chaque étage du micropipeline, avec la commande « `create_generated_clock` », qui permet à l'outil de récupérer les temps d'insertion de l'horloge globale. De cette manière, l'outil pourra effectuer une analyse temporelle entre deux horloges générées consécutives pour optimiser le chemin de donnée correspondant. D'après la Figure IV.8.b, nous obtenons l'Équation IV-3, où R_L correspond à la propagation de la requête, d au délai sur le chemin de la requête, G_L et G'_L représentent le temps de propagation depuis les contrôleurs vers les registres du premier et deuxième étage, T_L est le temps d'établissement des registres et enfin D_L est le temps maximum de traitement de la donnée dans la logique combinatoire.

$$R_L + d + G'_L - T_L > R_L + G_L + D_L \quad \text{Équation IV-3}$$

Considérons que la propagation depuis les contrôleurs jusqu'aux registres est équilibrée entre tous les étages du circuit, nous avons alors $G_L = G'_L$. La simplification de R_L , G_L et G'_L dans l'Équation IV-3 nous permet d'obtenir la valeur maximale de D_L grâce à l'Équation IV-4. Cette valeur est extraite pour chaque étage du micropipeline.

$$D_L < d - T_L$$

Équation IV-4

Comme le délai d est sur le chemin de l'horloge, l'outil de synthèse ne va pas chercher à changer sa valeur. Au moment de la synthèse, d vaut le temps de propagation d'une porte tampon. Cela signifie que la contrainte sur le chemin de données est très forte car d'après l'Équation IV-4, D_L doit être inférieur à la latence d'une porte tampon. Cette valeur ne pourra pas être atteinte mais l'outil fera le maximum pour optimiser le chemin de données.

L'analyse du réseau de contrôle, présentée dans la partie IV.4.1 nous permet de connaître tous les contrôleurs du circuit et de cette manière, tous les endroits où définir les horloges générées. Cela permet d'automatiser la définition des horloges du circuit et d'éviter au concepteur de devoir les décrire manuellement dans un langage interprétable par l'outil Design Compiler®.

Dans notre circuit d'exemple (Figure IV.2), vingt-et-un arcs sont coupés dans le réseau de contrôle, quatre pour chaque contrôleur, trois pour la fourche et deux pour la jonction. Une horloge globale et quatre horloges générées sont créées, comme le montre la Figure IV.9. Le signal d'acquittement a été enlevé car il est coupé.

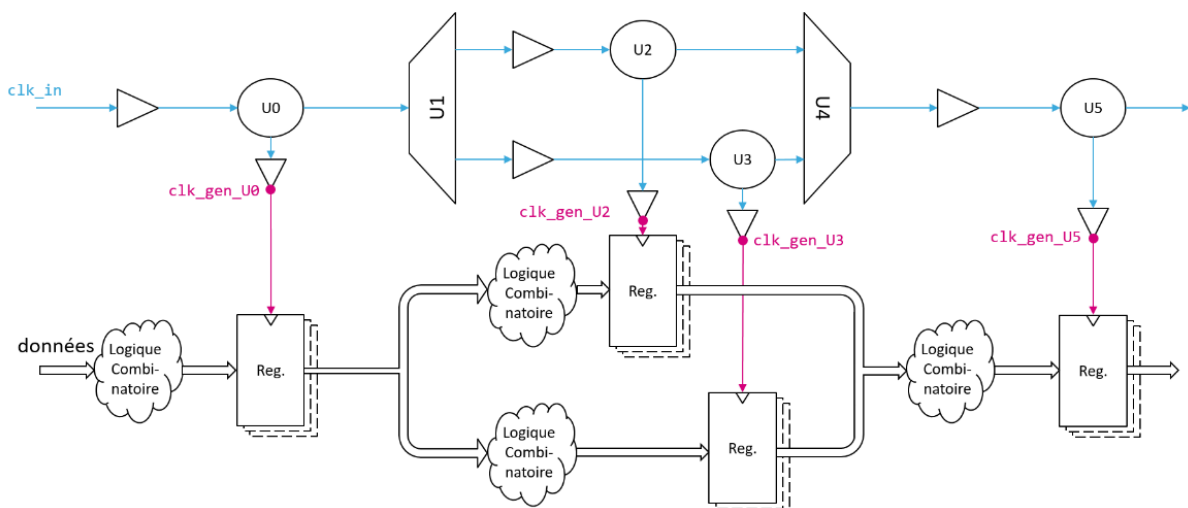


Figure IV.9. Définition des horloges dans le circuit d'exemple, l'horloge globale d'entrée se propage en bleu et les horloges générées en rose.

Maintenant que les boucles combinatoires sont coupées et que les horloges du circuit sont définies, l'outil peut effectuer la synthèse logique pour optimiser le chemin de données et retourner une description au niveau portes de notre circuit.

IV. 5. Analyse temporelle statique

Dans les circuits synchrones, cette étape du flot de conception permet de vérifier que l'étape de synthèse s'est bien déroulée et qu'il n'y a pas de violations temporelles dans le

circuit avec la période d'horloge définie par le concepteur. Pour cela, il peut extraire des rapports temporels spécifiques, par exemple, sur les chemins les plus critiques.

L'analyse temporelle statique (*Static Timing Analysis, STA*) repose sur le calcul et l'addition des délais de chaque porte du circuit. Elle permet de vérifier qu'il n'y a pas de violations temporelles, c'est-à-dire que les données sont stables lorsqu'elles sont mémorisées. Nous souhaitons éviter deux types d'erreurs illustrées sur la Figure IV.10. Premièrement, les erreurs d'établissement de la donnée (*setup* en anglais), causées par un signal trop lent. La donnée doit être stable suffisamment longtemps avant le front montant de l'horloge pour être mémorisée. Ce type d'erreur est illustré en bleu sur la Figure IV.10. Deuxièmement, les problèmes de maintien de la donnée (*hold* en anglais) qui sont causés par un signal trop rapide. La donnée change trop vite après le front montant de l'horloge donc c'est la donnée suivante qui est mémorisée et non celle voulue, en rose sur la Figure IV.10. L'étape d'analyse temporelle statique est essentielle pour que le circuit fonctionne.

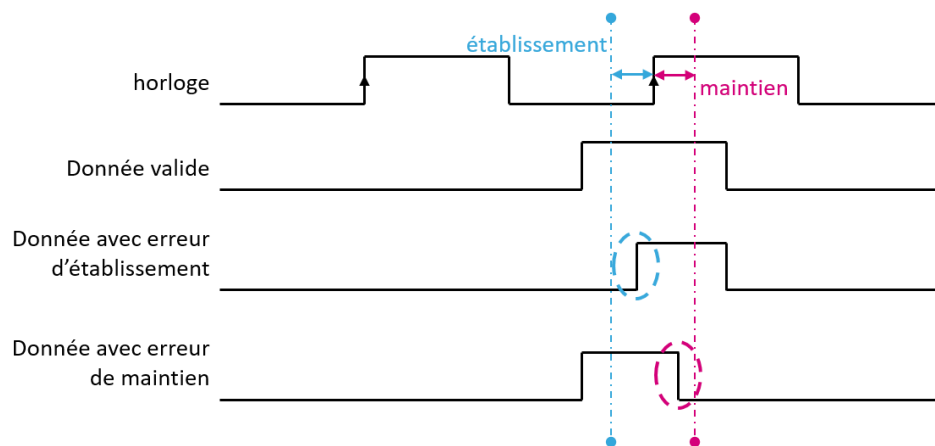


Figure IV.10. Illustration des problèmes d'établissement et de maintien de la donnée.

Dans les circuits synchrones, les problèmes d'établissement et de maintien des données se trouvent au niveau du front montant du signal d'horloge. En asynchrone, c'est au niveau du front montant de la requête que ces temps doivent être respectés. Il est nécessaire que la donnée soit stable suffisamment longtemps avant et après l'instant où le registre mémorise, donc l'instant où la requête arrive.

Comme pour les circuits synchrones le point d'entrée de l'analyse temporelle statique est la description au niveau portes du circuit ainsi que le jeu de contraintes temporelles du circuit. Cette description est obtenue en sortie de la synthèse logique de la partie précédente. Les contraintes de la synthèse sont réutilisées et enrichies pour pouvoir analyser tous les chemins du circuit. En asynchrone, nous ne cherchons pas à définir la fréquence de fonctionnement du circuit mais à équilibrer le chemin de la requête avec la logique combinatoire. Les valeurs de délais nécessaires pour la construction des pics de courant de notre modèle expliqué dans le chapitre III, sont extraites. Les valeurs minimales à insérer sur le chemin de la requête sont également définies au cours de cette étape. Le moteur d'analyse temporelle statique de l'outil

Design Compiler® de Synopsys est utilisé pour cette étape pour éviter d'avoir à changer d'outil.

IV. 5. 1. Analyse des chemins

Les contraintes définies lors de l'étape de synthèse logique sont réutilisées. Les boucles combinatoires sont coupées et les horloges sont définies pour chaque contrôleur. Cependant, plusieurs analyses sont nécessaires afin d'étudier les chemins qui rebouclent et pas seulement le chemin direct. Pour cela, une première analyse est effectuée pour extraire les valeurs du chemin direct. Par exemple, dans la Figure IV.11, la porte tampon bleue ciel de la fourche est coupée pour étudier le chemin bleu ciel. Dans les contraintes de synthèse, cette porte était coupée pour optimiser le chemin direct de la donnée. Une fois les informations extraites de ce chemin, l'arc est rétabli et l'autre sortie de la fourche est coupée pour étudier le second chemin. Sur la Figure IV.11, l'arc en rose est étudié en rétablissant le porte tampon bleue et en coupant la rose.

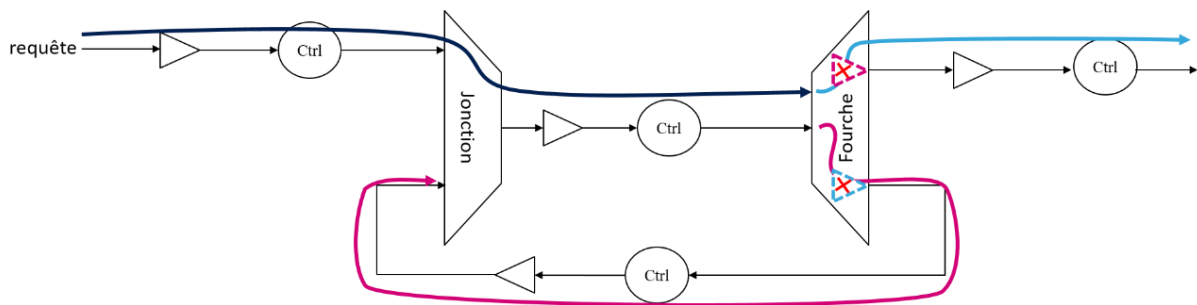


Figure IV.11. Exemple de boucle combinatoire du signal de requête.

Pour chaque jeu de contraintes de l'analyse temporelle statique, les informations de délais des chemins sont récupérées. De cette manière, toutes les valeurs du circuit sont extraites. La méthode de récupération des délais est décrite dans la sous-partie suivante.

Suite à cette étude, sur l'utilisation des outils synchrones pour réaliser la synthèse logique et l'analyse temporelle statique des circuits asynchrones, des travaux plus poussés ont été réalisés [68] pour analyser différents contrôleurs ainsi que le temps de maintien de la donnée. Dans notre cas, un circuit micropipeline utilisant un protocole quatre phases assure par construction que le délai nécessaire à l'étage précédent pour envoyer une nouvelle donnée est supérieur au temps de maintien des registres. La Figure IV.12 illustre cette hypothèse. Pour que l'étage 0 envoie une nouvelle donnée, donc que son signal de requête repasse au niveau logique haut, il faut que le protocole de la donnée précédente se termine, c'est-à-dire que l'étage 1 l'acquiesce. Pour cela, le signal d'acquiescement doit passer dans quatre portes, chemin représenté en rose sur la Figure IV.12.a, pour que la requête 0 passe au niveau logique bas. Elle doit ensuite se propager jusqu'au contrôleur suivant pour que l'acquiescement de l'étage 1 vers l'étage 0 repasse au niveau bas et que le protocole se termine. Cette propagation est

Avec l'outil Design Compiler® nous pouvons obtenir des rapports temporels entre deux horloges qui ont un lien. Par exemple, si nous voulons obtenir le rapport entre l'étage 0 et l'étage 2, donc entre l'horloge « `clk_gen_U0` » et « `clk_gen_U2` », définies dans la Figure IV.9, nous utilisons la commande « `get_timing_path -from clk_gen_U0 -to clk_gen_U2` ». L'outil nous retourne différentes valeurs concernant ce chemin. Nous nous intéressons à la différence entre le délai de la propagation de la requête et celui de la propagation de la donnée. Cette valeur est appelée « *slack* ». Si elle est négative, la requête est trop rapide et il est nécessaire d'ajouter du délai. Sinon la donnée ne sera pas correctement mémorisée. Dans notre cas, lors de l'étape de la conception de notre circuit nous n'avons pas ajouté de délais entre les contrôleurs mais seulement une porte tampon. Il est donc normal d'avoir des valeurs négatives. Celles-ci vont nous servir à déterminer les valeurs minimales des délais à insérer. Pour calculer cette valeur, nous avons besoin du délai déjà présent pour traverser le contrôleur et la porte tampon, valeur disponible avec la commande précédente, ainsi que de la valeur manquante pour mémoriser la bonne donnée, le *slack*.

Grâce à l'analyse de réseau et à la définition automatique des horloges du circuit, nous avons également pu automatiser l'extraction des délais des différents chemins. De cette manière, tous les chemins sont analysés et les informations de chacun sont retournées dans un fichier. La Figure IV.13 montre le fichier avec les valeurs de délais de notre exemple. L'horloge de départ et celle d'arrivée sont spécifiées pour savoir quel chemin est concerné, ensuite les valeurs du *slack* et du délai de propagation de la requête sont indiquées. Nous remarquons une autre information nommée « *ctrl2reg* », elle correspond au délai entre le contrôleur et les registres qu'il contrôle. Celle-ci est identique pour tous les chemins car tous les registres doivent être équilibrés entre eux. Cette valeur est une contrainte qui sera spécifiée par le concepteur, dans notre cas elle est fixée à 0,3 ns. Enfin, la dernière valeur de chaque chemin est celle du délai minimum à insérer, *delai_min*. Elle est calculée à partir de l'Équation IV-6.

$$delai_min = |slack| + latence_req \quad \text{Équation IV-6}$$

expediteur	delay evaluation destination	slack	latence_req	ctrl2reg	delai_min
v clk_in	clk_gen_U0	-0.812	0.688	0.3	1.5
clk_gen_U0	clk_gen_U3	-0.326	0.641	0.3	0.967
clk_gen_U3	clk_gen_U5	-0.239	0.954	0.3	1.193
clk_gen_U0	clk_gen_U2	-0.333	0.641	0.3	0.974
clk_gen_U2	clk_gen_U5	-0.243	0.954	0.3	1.197

Figure IV.13. Extraction des valeurs des délais du circuit d'exemple.

Nous récupérons également pour chaque chemin le nombre de registres contrôlés par l'horloge et la valeur du chemin critique. Toutes ces informations sont également mémorisées dans un fichier Matlab®. Cela, nous permet de définir la forme des pics de consommation de courant pour chaque étage, utilisée dans l'étape de co-simulation.

IV. 6. Co-Simulation

Maintenant que nous avons récupéré toutes les informations de délais nécessaires, nous allons pouvoir passer à l'étape de co-simulation. Celle-ci a pour but de trouver les valeurs finales à insérer pour que le rayonnement électromagnétique du circuit soit en dessous du gabarit défini. Pour cela, l'algorithme génétique qui crée et évalue des combinaisons de délais est couplé avec le simulateur analogique rapide, qui évalue la consommation de courant du circuit. Cette étape du flot de conception, nouvelle par rapport à un flot de conception classique, est néanmoins réalisée avec des outils standards.

IV. 6. 1. Algorithme Génétique

L'algorithme génétique a pour but de trouver les délais à insérer dans le circuit pour que son spectre respecte un gabarit. Pour cela, il crée des populations d'individus, candidats au problème, composées de gènes, qui correspondent aux paramètres.

Dans notre cas, une population est composée de cinquante individus et leurs gènes correspondent aux délais à insérer sur le chemin du signal de requête. Dans le circuit d'exemple de ce chapitre (Figure IV.2), chaque individu possède quatre gènes. La valeur de chacun est comprise entre le délai minimum à insérer, extrait au cours de l'analyse temporelle statique, et la valeur maximale définie par le concepteur. Cette dernière peut être limitée par les spécifications du circuit, la latence maximale du circuit, ou la place physique disponible. Le pas entre les valeurs disponibles pour les délais dépend de la latence de l'élément de délai de base qui sera inséré sur le chemin de la requête.

Dans notre exemple, notre élément de délai de base est une porte tampon qui possède une latence de 0,06 ns. Nous avons décidé de donner onze valeurs possibles pour chaque délai. La valeur maximale est définie par l'Équation IV-7.

$$d_{max} = d_{min} + 10 * (0,06 * 10^{-9}) \quad \text{Équation IV-7}$$

Pour générer les individus d'une population, l'algorithme va piocher des valeurs parmi celles disponibles pour les quatre gènes et créer des combinaisons. Ensuite, pour envoyer ces combinaisons à tester au simulateur analogique rapide, l'algorithme génétique crée pour chaque individu un fichier standard de délais (*Standard Delay Format, SDF*) avec ses gènes pour les cellules de délais du chemin de requête.

La Figure IV.14 montre la description de la première cellule de délai de notre circuit d'exemple, dans le fichier standard de délais. Les triplets de délais ont la même valeur. Nous ne cherchons pas à faire des simulations en fonction du processus de fabrication du circuit mais seulement à obtenir la courbe en courant pour la combinaison de délais de l'algorithme génétique. Le premier triplet donne le délai pour la propagation du front montant et le deuxième pour celle du front descendant.

```

(CELL
  (CELLTYPE "rds_delay")
  (INSTANCE rds_exemple_model/delay_stagel)
  (DELAY
    (ABSOLUTE
      (IOPATH s_req m_req (1.6200::1.6200::1.6200) (1.4520::1.4520::1.4520))
      (IOPATH s_ack m_ack (0.4700::0.4700::0.4700) (0.5800::0.5800::0.5800))
    )
  )
)

```

Figure IV.14. Exemple de la description d'une cellule de délai dans le fichier standard de délais (.sdf) envoyé au simulateur analogique rapide.

IV. 6. 2. Simulation Analogique Rapide

Le fichier SDF fourni par l'algorithme génétique permet d'annoter le réseau de Petri temporel du circuit. Ainsi, le simulateur analogique rapide est capable de simuler le comportement du circuit avec les délais spécifiques à un individu. La simulation permet d'extraire les temps de commutation des différents étages d'un individu. L'instant où le front descendant du signal de requête arrive sur l'entrée des registres est également enregistré. Cette commutation engendre un petit pic de courant. Ces valeurs sont mémorisées dans un fichier Matlab®.

La deuxième étape pour obtenir la consommation totale du circuit est d'évaluer les pics de courant de chaque étage. La forme des pics de courant ne dépend pas du délai inséré entre les contrôleurs, elle est propre au circuit. Les paramètres de chaque pic sont préalablement calculés à l'aide des valeurs extraites au cours de l'analyse temporelle statique. Nous pouvons ensuite associer à chaque temps de commutation son pic de courant. Cette étape est réalisée grâce à une fonction développée sous Matlab®. Cette courbe est ensuite renvoyée à l'algorithme génétique pour qu'il évalue le potentiel de la combinaison de délais par rapport au gabarit.

Cette étape de modélisation du courant est très importante car en fonction du protocole de communication utilisé et de la structure du circuit il peut être très difficile de calculer les instants de commutation des registres. En effet, dans un circuit non-linéaire, l'ordre de commutation de certains pics peut être inversé, nous verrons que c'est le cas dans le circuit d'exemple utilisé. De plus, il est nécessaire de prendre en compte le délai du signal d'acquittement dans le protocole pour calculer les temps de propagation du signal de requête. C'est pourquoi, la simulation du réseau de Petri temporel permet de faciliter l'obtention de la consommation de courant du circuit.

La Figure IV.15 montre la consommation de courant de deux individus de l'algorithme génétique. Les pics de courant des étages U2 et U3 (représentés sur la Figure IV.5) sont mis en avant respectivement en bleu et vert. Ces deux étages sont concurrents et ils possèdent chacun leur délai. Suivant la valeur de ces délais, les pics n'arrivent pas dans le même ordre. L'individu de la Figure IV.15.a possède un délai d_2 inférieur à d_3 ainsi le pic de l'étage U2, en

bleu, arrive avant celui de l'étage U3, en vert. En revanche, l'inverse se produit pour l'individu de la Figure IV.15.b qui possède un délai d_2 supérieur à d_3 . La simulation du circuit permet également de prendre en compte le temps attendu dans la porte de jonction, pour que les deux requêtes soient disponibles en même temps. C'est pourquoi, le deuxième pic rouge n'arrive pas en même temps sur les deux simulations. Les pics avec une amplitude inférieure correspondent aux fronts descendants du signal de requête. Nous pouvons constater que celui du premier étage arrive pendant le pic du front montant du quatrième étage.

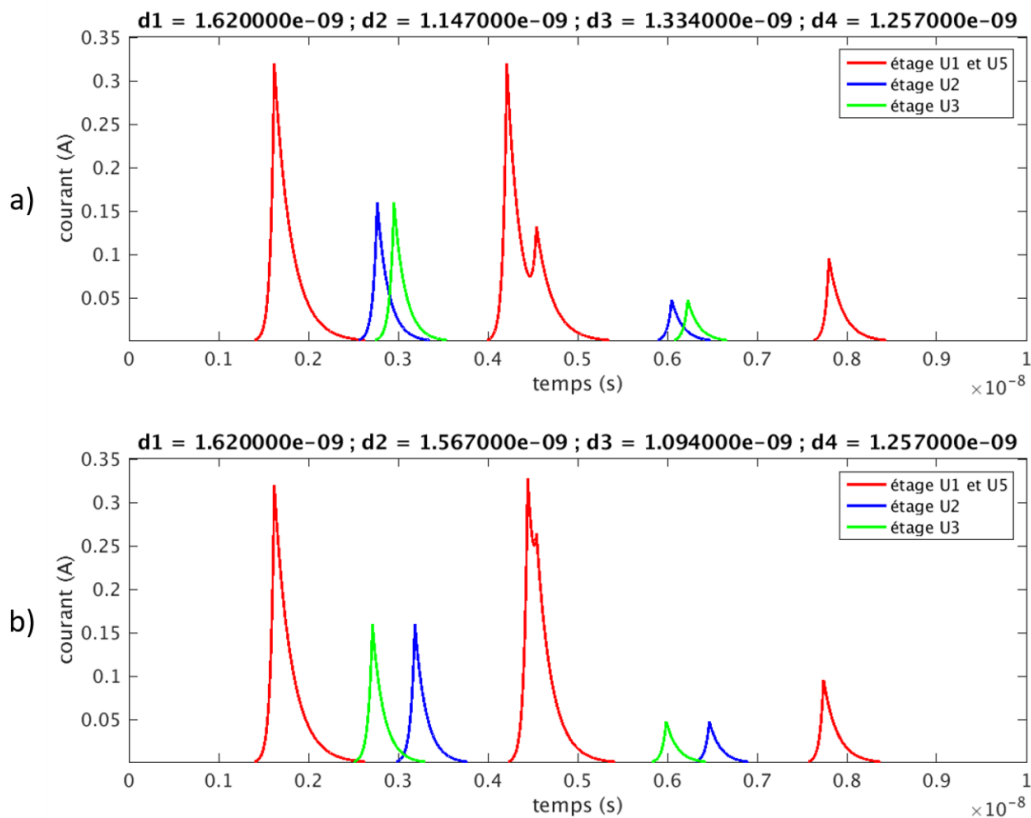


Figure IV.15. Consommation de courant, des différents étages, pour deux individus différents.

La Figure IV.15 montre la consommation de courant pour la propagation d'une seule donnée dans le circuit. Cependant, pour effectuer la transformée de Fourier, nous prenons en compte la propagation de plusieurs données à la suite. Le délai entre l'envoi de deux données peut être choisi par le concepteur en fonction de l'application du circuit. Il est également possible d'envoyer une nouvelle donnée avant la fin du traitement de la première. Dans ce cas-là, le comportement est simulé dans le fichier de test du réseau de Petri temporel afin d'extraire les temps de commutation de la deuxième donnée, qui dépendent de la propagation de la première.

IV. 6. 3. Obtention d'une solution

La dernière étape de la co-simulation est réalisée par l'algorithme génétique. Celui-ci calcule la transformée de Fourier de la consommation de courant de l'individu pour comparer

son spectre avec le gabarit défini par l'utilisateur. Cette étape permet de calculer la note de l'individu. Elle représente sa capacité à résoudre le problème. Cet enchaînement d'opérations et d'échanges entre l'algorithme génétique et le simulateur analogique rapide est effectué pour chaque individu de la population. Une fois tous les individus évalués, si la note vaut 0 alors une solution est trouvée et le processus s'arrête. Sinon, l'algorithme génétique va créer une nouvelle génération d'individus et la co-simulation recommencera tant qu'une solution n'est pas trouvée.

Lorsque l'algorithme génétique a trouvé une solution il renvoie les délais entre chaque pic de courant. Pour obtenir les délais à insérer sur le chemin de contrôle, il faut soustraire à cette valeur le délai déjà présent dans le chemin de contrôle, par exemple le temps nécessaire pour traverser un contrôleur.

Pour trouver une solution, pour les quatre délais de notre circuit d'exemple, neuf générations ont été évaluées au cours de la co-simulation. La Figure IV.16 montre le spectre électromagnétique obtenu pour la combinaison finale (courbe en bleue). Nous pouvons constater que le spectre respecte bien le gabarit (courbe en rouge) défini dans la première étape du flot de conception, dans la partie IV.3.

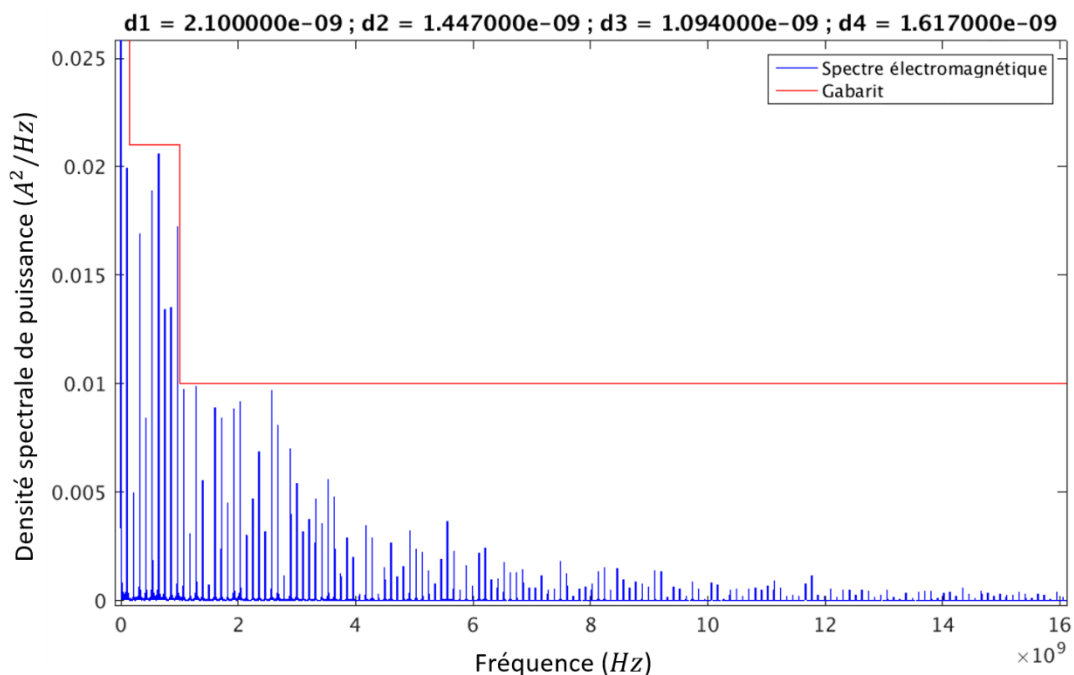


Figure IV.16. Spectre électromagnétique, courbe bleue, de la solution trouvée par la co-simulation, celui-ci est bien en dessous du gabarit, courbe rouge.

Une fois cette étape de mise en forme du spectre électromagnétique finie, nous connaissons les valeurs des délais à insérer dans le circuit. Une analyse temporelle statique finale est à nouveau effectuée en annotant les portes tampons, déjà présentes entre les contrôleurs, avec les valeurs obtenues. Ainsi, le concepteur peut s'assurer qu'il n'y a pas de violations temporelles dans le circuit avec les valeurs de délais finales. Enfin, le circuit peut être implémenté en utilisant la même étape qu'un flot de conception classique.

IV. 7. Implémentation

IV. 7. 1. Implémentation synchrone

L'implémentation physique d'un circuit synchrone est généralement divisée en trois parties, la définition des zones du circuit, le placement des cellules et le routage des connexions.

Dans un premier temps, la surface du circuit est définie ainsi que les zones pour chaque partie du circuit, cette étape est appelée la définition du « *floorplan* ». Dans la deuxième étape, les cellules du circuit sont placées dans les zones définies. Une optimisation du placement des cellules est réalisée. Ensuite l'arbre d'horloge est construit et ses cellules sont placées. Dans la dernière étape, le routage des connexions du circuit est réalisé. Le routage de l'arbre d'horloge est d'abord effectué pour équilibrer tous les chemins. Ensuite le chemin de données est routé.

Notre implémentation étant régie par les contraintes temporelles du circuit, nous pouvons découper un peu différemment le flot après l'étape de « *floorplan* ». Nous centrons ces étapes autour de la construction de l'arbre d'horloge, ce qui nous donne :

- L'optimisation avant la création de l'arbre d'horloge.
- L'élaboration de l'arbre d'horloge.
- L'optimisation après la création de l'arbre d'horloge.

L'optimisation du placement des cellules du circuit est réalisée avant la création de l'arbre d'horloge. Ensuite, l'outil détermine combien de cellules sont nécessaires pour créer l'arbre d'horloge. Elles sont ensuite placées et routées pour que les chemins soient équilibrés. L'optimisation, après la création de l'arbre d'horloge, consiste à router les connexions du circuit. Puis l'arbre d'horloge peut être rééquilibré en fonction des délais dans la logique combinatoire.

IV. 7. 2. Implémentation asynchrone

Dans notre cas, nous allons utiliser les mêmes étapes que dans un flot classique. En synchrone l'implémentation est réalisée autour du signal d'horloge. En asynchrone, nous allons nous concentrer sur le signal de requête, comme le montre la représentation des trois étapes sur la Figure IV.17. Avant la création de l'arbre de requête, nous allons optimiser le chemin de données et placer les cellules dans les zones préalablement définies. Lors de l'étape de création de l'arbre de requête, les chemins des contrôleurs vers les éléments de mémorisation sont équilibrés (deuxième étape de la Figure IV.17). Dans la dernière étape, les délais sont insérés sur le chemin de la requête entre les contrôleurs pour obtenir les valeurs trouvées dans l'étape de co-simulation.

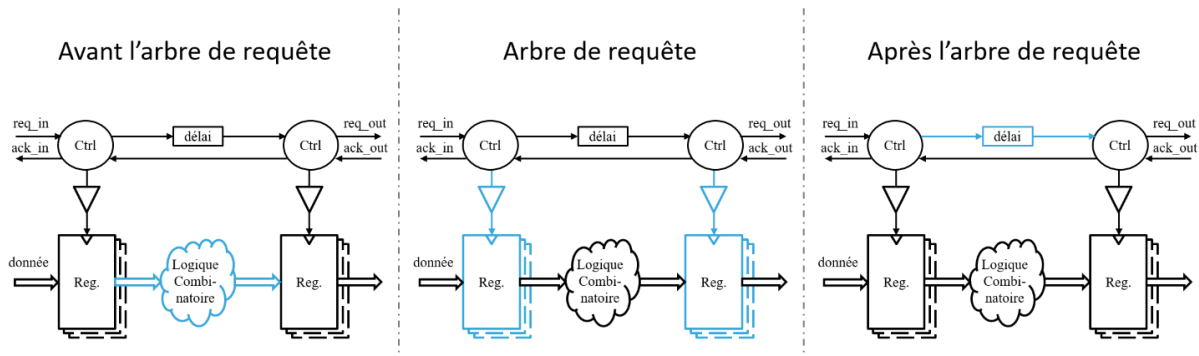


Figure IV.17. Etapes de l'implémentation d'un circuit micropipeline dans notre flot de conception.

Pour que les valeurs de délais trouvées soient respectées une fois le circuit fabriqué, il est nécessaire de prendre en compte les variations des procédés de fabrication. Pour cela, deux stratégies peuvent être adoptées. La première solution consiste en l'analyse du circuit pour les différents cas de fonctionnement post fabrication : meilleur, typique ou pire cas. Une combinaison de délais est trouvée pour chaque condition. Ensuite, en ajoutant une ligne de délais programmables entre les contrôleurs, il sera possible de choisir la combinaison qui correspond au fonctionnement du circuit une fois fabriqué. La deuxième solution consiste à réaliser l'analyse du circuit seulement dans le cas de fonctionnement le plus lent. Une seule solution est trouvée dans ces conditions pour les délais. Une ligne de délais programmables est également ajoutée entre les contrôleurs de sorte qu'après fabrication, si le circuit a un fonctionnement plus rapide, il est toujours possible d'ajouter un délai supplémentaire afin de retrouver les valeurs trouvées initialement pour les délais.

IV. 8. Conclusion

Ce chapitre présente notre flot de conception qui permet de contrôler le rayonnement électromagnétique des circuits numériques. Toutes les étapes du flot sont réalisées avec les outils de CAO standards utilisés pour les circuits synchrones. La première partie donne une vue d'ensemble des étapes qui le compose et de leurs enchaînements.

Ensuite, chaque étape est détaillée et illustrée à l'aide d'un circuit d'exemple. La première étape correspond à la conception du circuit en utilisant les circuits micropipelines. Les chemins de données et de contrôle sont décrits dans un langage matériel au niveau registres. Les délais à insérer sur le chemin du signal de requête ne sont pas encore connus, ils ne sont donc pas insérés à ce stade, seule une porte tampon est ajoutée. C'est également dans cette étape que le gabarit fréquentiel à respecter est défini par le concepteur.

La deuxième étape du flot est la synthèse logique du circuit. La description en langage matériel est convertie en portes logiques de la bibliothèque spécifiée. Pour cela, une analyse du réseau de contrôle permet de récupérer les informations sur les éléments de contrôle présents dans le circuit. Une image du réseau est renvoyée au concepteur pour lui permettre

de vérifier visuellement son réseau de contrôle. Toutes les informations sur les contrôleurs et les chemins à analyser sont stockées dans un tableau. Elles permettent ensuite de connaître les arcs à couper pour enlever toutes les boucles combinatoires du réseau de contrôle. Le signal d'acquiescement doit être coupé pour éviter les boucles avec le signal de requête. De plus, si la porte de Muller est conçue avec des portes logiques standards, elle possède également des boucles combinatoires. Une horloge est ensuite définie sur le port d'entrée de la requête. Pour pouvoir analyser les différents chemins entre les contrôleurs, des horloges générées en sortie des contrôleurs sont définies. Ces contraintes permettent à l'outil d'optimiser le chemin de données du circuit pour, ensuite, renvoyer sa description avec des portes logiques.

La partie suivante décrit l'analyse temporelle statique du circuit. Elle permet d'analyser les délais dans le circuit à partir de la description au niveau portes et des contraintes temporelles. Ces deux paramètres sont extraits de la synthèse logique. Néanmoins, les contraintes doivent être enrichies pour analyser tous les chemins du circuit. Celles prédéfinies en synthèse nous permettent d'analyser le chemin direct de la donnée. Cependant, elles ne permettent pas de récupérer toutes les informations si la requête reboucle dans le circuit. Pour extraire les délais de la boucle, il est nécessaire de définir de nouvelles contraintes pour couper le chemin direct et rétablir la boucle de la requête. Une nouvelle analyse est réalisée avec ces nouvelles contraintes. Les rapports temporels permettent de récupérer les valeurs de délais nécessaires pour construire les pics de courant et définir les délais minimaux à insérer sur le chemin de la requête. Toutes ces étapes sont automatisées avec des scripts et des contraintes standards en *TCL* qui sont supportés par la majorité des outils de CAO, utilisés dans un flot classique.

L'étape présentée dans la partie suivante permet de mettre en forme le rayonnement électromagnétique du circuit. Pour cela, une co-simulation est créée entre l'algorithme génétique et notre simulateur analogique rapide. L'algorithme commence par définir des individus qui représentent des combinaisons de délais pour le chemin de contrôle. Il crée pour chacun un fichier standard de délais qui permet d'annoter le réseau de Petri temporel avec les délais des individus. Ce fichier est ensuite envoyé au simulateur. Celui-ci permet de simuler le réseau de Petri temporel avec les valeurs spécifiques à un individu et d'extraire les temps de commutation de chaque étage. Le simulateur associe à chaque temps de commutation le pic de courant qui correspond à l'étage. Ce pic est déterminé grâce aux délais extraits lors de l'analyse temporelle statique comme expliqué dans le chapitre III. Cette association permet d'obtenir la consommation de courant du circuit, qui est envoyée à l'algorithme génétique. Après avoir calculé la transformée de Fourier de la courbe de courant, l'algorithme compare le spectre avec le gabarit défini par le concepteur. Une fois que le spectre d'un individu est en tout point en dessous du gabarit, ses valeurs sont utilisées pour l'implémentation du circuit.

La dernière partie décrit notre méthode d'implémentation du circuit micropipeline. Celle-ci est effectuée de la même manière que pour un circuit synchrone mais par rapport aux signaux de requête et non au signal d'horloge. Dans un premier temps, la surface et les différentes zones du circuit sont définies. Dans l'étape suivante, les cellules du circuit, à

l'exception des délais du chemin de requête, sont placées. Les arbres de requête sont alors construits afin que les chemins entre les contrôleurs et les éléments de mémorisation soient équilibrés. Enfin, les délais du chemin de requête sont insérés entre les contrôleurs pour retrouver les valeurs définies par la co-simulation. Des lignes de délais programmables peuvent être ajoutées sur les chemins de requête pour ajuster les délais après fabrication et revenir aux valeurs de délais désirées.

Notre flot de conception a été évalué avec la réalisation de circuits de test qui sont présentés dans le chapitre suivant. Un premier circuit a été fabriqué sur silicium et un second est prêt à être fabriqué.

Chapitre V.

Circuits de test et mesures

V. 1.	Introduction.....	88
V. 2.	Paramètres pour aider à contrôler le rayonnement.....	89
V. 2. 1.	Concevoir le circuit en asynchrone	89
V. 2. 2.	Résolution des délais.....	91
V. 2. 3.	Changement d'architecture.....	93
V. 3.	Circuit de test « Pegase »	94
V. 3. 1.	Architecture du circuit.....	95
V. 3. 2.	Simulations.....	96
V. 3. 3.	Réalisation du circuit	97
V. 4.	Mesures.....	98
V. 4. 1.	Temporelle.....	99
V. 4. 2.	Courant.....	100
V. 4. 3.	Electromagnétique	102
V. 5.	Circuit de test « Hydra »	111
V. 5. 1.	Architecture du circuit.....	112
V. 5. 2.	Simulation	113
V. 5. 3.	Réalisation du circuit	114
V. 5. 4.	Simulation après implémentation	116
V. 6.	Conclusion	117

Résumé :

Certains paramètres tels que la résolution des délais possibles ou l'architecture du circuit peuvent être modifiés dans le circuit pour atteindre les spécifications électromagnétiques visées.

Notre méthode a été validée à l'aide de mesures électromagnétiques réalisées sur un circuit de test conçu lors de ma première année de thèse. Différentes combinaisons de délais ont été mesurées, grâce à des délais programmables entre les étages de contrôle, pour montrer que le rayonnement est bien mis en forme afin d'éviter des plages de fréquences spécifiques.

V. 1. Introduction

Ce chapitre a pour objectif de présenter les circuits de test qui nous ont permis de mettre en place et de tester le flot de conception décrit dans le chapitre IV.

La première partie de ce chapitre présente les paramètres que le concepteur peut modifier dans son circuit pour obtenir un spectre électromagnétique en dessous du gabarit. Nous verrons dans un premier temps que, comparé à une technique de désynchronisation, une conception asynchrone permet de trouver une solution plus rapidement. Dans un circuit asynchrone les différents étages combinatoires peuvent avoir des délais très différents. Cela nous permet de mieux mettre en forme le courant donc, le spectre. Dans un deuxième temps, l'impact de la résolution du pas entre les délais possibles est présenté. Une granularité plus fine permet d'éviter des pics dans le spectre. Enfin, nous verrons que des changements dans l'architecture du circuit peuvent aider la co-simulation à mettre en forme le spectre électromagnétique. Diviser certains étages permet d'augmenter le nombre de paramètres pour déplacer les pics de courant en gardant la même consommation totale pour les registres.

La deuxième partie présente notre premier circuit de test. Un circuit de cryptographie de norme de chiffrement avancé (*Advanced Encryption Standard, AES*) a été conçu en asynchrone. Tout d'abord, l'architecture est décrite. Nous verrons que ce circuit est non-linéaire, avec une boucle du signal de requête. Des lignes de délais programmables ont été ajoutées entre les contrôleurs. Elles permettent de changer le temps de calcul de l'AES et de déplacer les appels de courant des étages. De cette manière, plusieurs combinaisons de délais pourront être mesurées. Un signal de test du fonctionnement de l'AES a été créé. Celui-ci passe au niveau logique haut lorsque l'AES commence à calculer et au niveau logique bas à la fin du calcul. Une simulation logique est présentée pour montrer les différents signaux dont celui de test. Enfin, la réalisation du circuit est présentée. L'AES a été implémenté, en technologie CMOS 40 nm STMicroelectronics, dans un circuit de test effectué par l'équipe que j'ai intégrée à STMicroelectronics. Les problèmes rencontrés au cours de l'implémentation physique du circuit nous ont permis de mieux définir les contraintes à appliquer.

Ensuite, les mesures de notre premier circuit sont présentées. Tout d'abord, nous avons réalisé des mesures temporelles pour vérifier le fonctionnement de notre circuit. La mesure du signal de test avec un oscilloscope nous a permis de vérifier que le circuit fonctionne et de mesurer le temps nécessaire pour crypter le message en fonction des valeurs des délais programmables. Ensuite, des mesures de courant ont été réalisées au niveau d'un port du circuit créé pour obtenir le courant. Elles nous ont permis de voir les changements sur le courant lorsque le circuit fonctionne. Enfin, nous avons effectué des mesures de rayonnement avec différentes méthodes. Nos premiers essais ont été réalisés dans une cage et dans une mini cage de Faraday avec une antenne reliée à l'analyseur de spectre. Des mesures avec une cellule de mesure Electromagnétique Transverse (*Transverse ElectroMagnetic, TEM*) ont également été réalisées. Nous avons utilisé un scanner de surface pour se placer précisément

au-dessus du circuit et récupérer le spectre avec une antenne construite avec une boucle de 150 μm de diamètre.

Enfin, notre deuxième circuit de test prêt à être fabriqué est présenté. L'architecture de celui-ci est plus simple pour pouvoir mieux corrélérer les mesures avec les simulations. Le circuit contient cinq étages avec le même chemin de données composé d'un OU Exclusif entre la sortie du registre précédent et un bit de polarité. La sortie du OU Exclusif est mémorisée dans l'étage suivant et ainsi de suite. Le bit de polarité permet de faire ou non commuter les registres. Quant au chemin de contrôle, nous garderons la même structure que pour le premier circuit de test. Les instants des fronts montants, des requêtes, calculés au cours d'une simulation au niveau portes sont comparés avec les instants des pics de courant de notre simulateur analogique rapide. Ensuite, la réalisation du circuit dans deux technologies différentes est présentée. Pour finir, des simulations après placement et routage, dans différentes conditions de fonctionnement et avec l'extraction des parasites sont illustrées.

V. 2. Paramètres pour aider à contrôler le rayonnement

Pour pouvoir atteindre les spécifications électromagnétiques le concepteur peut modifier certains paramètres du circuit. Les sous-parties qui suivent, présentent et illustrent à l'aide d'exemples, les effets de ces changements sur le spectre.

V. 2. 1. Concevoir le circuit en asynchrone

La similarité du chemin de données entre les circuits micropipelines et les circuits synchrones rend possible la désynchronisation de circuits existants. Pour cela, le chemin de données d'un circuit synchrone est conservé et seul le chemin de contrôle correspondant a besoin d'être conçu.

La désynchronisation d'un circuit permet de trouver une solution pour un gabarit donné. Cependant, un circuit synchrone est réalisé avec des contraintes temporelles fortes. Les différentes parties combinatoires sont équilibrées pour éviter le ralentissement de tout le circuit par un seul étage au moment de la définition de l'horloge. A cause de cette contrainte les valeurs minimales et maximales des délais de chaque étage désynchronisé seront proches pour ne pas augmenter la latence totale. En revanche, en conception asynchrone les étages peuvent être très déséquilibrés, ce qui permet d'avoir des valeurs de délais très différentes pour chaque étage. Cela permet à l'algorithme de trouver une solution plus facilement.

Un circuit micropipeline linéaire de cent étages a été modélisé pour comparer ses versions synchrone, désynchronisé et asynchrone. La version synchrone est approchée en utilisant un circuit micropipeline avec les mêmes délais pour tous les étages. Pour ce cas d'étude, une période de 10 ns est fixée. Pour la version désynchronisée les délais sont contraints entre 8 ns et 10 ns avec un pas de 0,1 ns. La limite supérieure permet de s'assurer que la latence totale

du circuit est maintenue durant l'optimisation. La borne inférieure correspond au chemin critique des étages. La Figure V.1 représente le spectre électromagnétique de la version synchrone, courbe verte, et de la version désynchronisée, courbe bleue. Pour le gabarit défini sur la Figure V.1, courbe noire, le circuit désynchronisé permet de trouver une solution pour que le rayonnement respecte les spécifications.

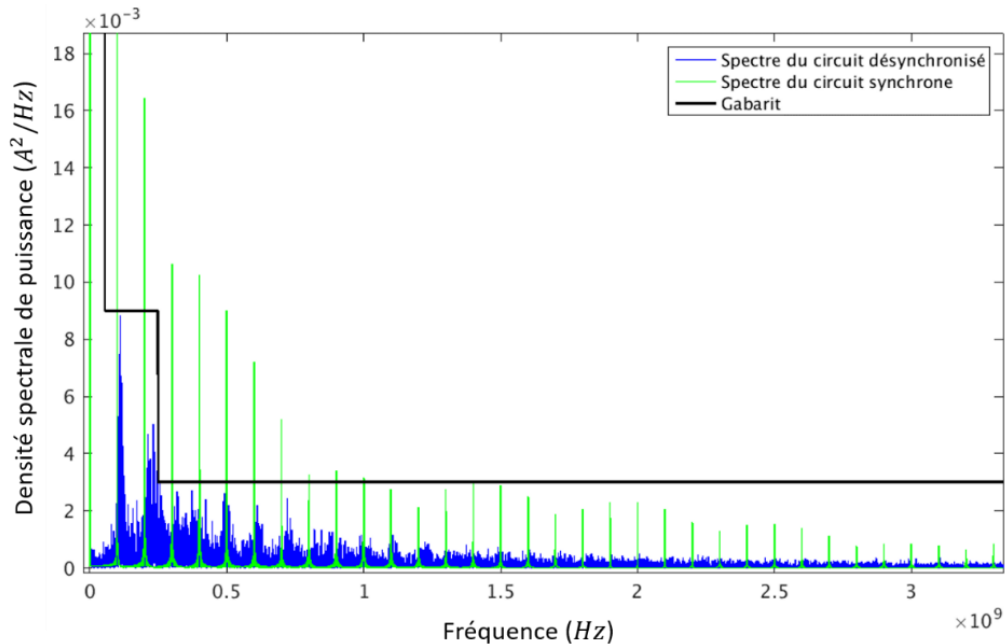


Figure V.1. Représentation du spectre électromagnétique du circuit synchrone avec une période de 10 ns, du circuit désynchronisé et du gabarit utilisé pour l'optimisation.

Cependant, la borne inférieure des délais, imposée par la version synchrone, peut devenir limitante pour une optimisation plus agressive, avec un gabarit plus contraint. Pour illustrer cette limitation, nous avons réutilisé le micropipeline de cent étages mais en diminuant la borne inférieure pour les délais. Dans la version asynchrone, les délais peuvent prendre une valeur entre 6 et 10 ns avec un pas de 0.1 ns.

Sur la Figure V.2, la courbe bleue représente le spectre du circuit désynchronisé après 134 générations pour l'algorithme génétique. Celui du circuit asynchrone, après 14 générations, est représenté en rouge. Le gabarit utilisé pour les deux optimisations est en noir. Nous constatons que la co-simulation a trouvé une solution pour le circuit asynchrone, alors qu'elle n'a pas trouvé de solution pour la version désynchronisée car des pics bleus dépassent du gabarit. Grâce aux différences de délais entre les chemins combinatoires le circuit asynchrone a une plage de délais plus importante. Il est donc plus facile pour la co-simulation de contrôler le rayonnement.

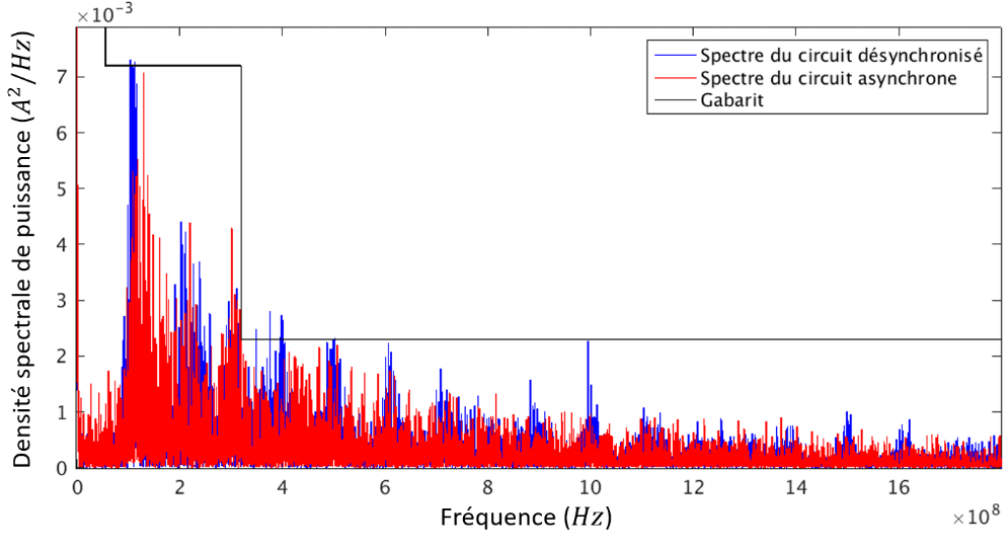


Figure V.2. Spectre du circuit désynchronisé après 134 générations de l'algorithme génétique comparé à la solution trouvée pour le circuit asynchrone après 14 générations.

Un circuit désynchronisé permet de contrôler le spectre électromagnétique en utilisant notre flot. Cependant, la plage de délais disponible peut être insuffisante si la réduction demandée est trop importante. Pour pallier à cette limitation, il est préférable de concevoir le circuit directement en logique asynchrone. Cela permet, d'avoir des valeurs de délais plus variées pour mieux répartir les pics de courant et contrôler le rayonnement.

V. 2. 2. Résolution des délais

Un autre paramètre important lors de l'étape de co-simulation est la résolution du pas des délais possibles pour contrôler le spectre. Notre modèle de courant est représenté par une somme de doubles exponentielles, notons D_{exp} notre fonction pour un pic de courant. Le passage temporel vers fréquentiel est exprimé dans l'Équation V-1, où D_p représente le délai entre l'étage p et l'entrée du circuit.

$$f(t) = \sum_{p=0}^n D_{exp}(t - D_p) \xleftrightarrow{FFT} F(f) = \sum_{p=0}^n D_{exp}(f) * e^{-2i\pi D_p f} \quad \text{Équation V-1}$$

Notons R la résolution du pas des délais dans l'intervalle possible. Nous avons l'Équation V-2 pour le délai D_p de l'étage p .

$$\forall D_p, \exists a_p \in \mathbb{N}, \text{ tel que } D_p = a_p * R \quad \text{Équation V-2}$$

Pour $f = \frac{k}{R}$ avec $k \in \mathbb{N}$ et l'expression de D_p de l'Équation V-2, nous obtenons l'Équation V-3.

$$F\left(\frac{k}{R}\right) = \sum_{p=0}^n D_{exp}\left(\frac{k}{R}\right) * e^{-2i\pi a_p k} \quad \text{Équation V-3}$$

Or, $a_p \in \mathbb{N}$ donc $e^{-2i\pi a_p k} = 1$, de cette manière, nous obtenons l'expression finale, du spectre électromagnétique, au niveau des fréquences inversement proportionnelles au pas de résolution avec l'Équation V-4.

$$F\left(\frac{k}{R}\right) = n * D_{exp}\left(\frac{k}{R}\right) \quad \text{Équation V-4}$$

Les termes D_p et a_p n'étant plus présents, nous remarquons que le spectre, à ces fréquences, est indépendant des délais. Les raies inversement proportionnelles à la résolution des délais ne peuvent pas être diminuées grâce aux délais. Il est donc important de choisir le pas en fonction du gabarit souhaité.

La Figure V.3 montre le spectre pour un circuit micropipeline linéaire de cent étages avec une résolution pour les délais de 1 ns, pour le spectre rouge, et de 0,5 ns pour le bleu. Les spectres présentent des harmoniques aux fréquences inversement proportionnelles à la résolution comme attendu d'après l'Équation V-4. Le spectre rouge a des raies tous les mégahertz et le bleu tous les deux mégahertz.

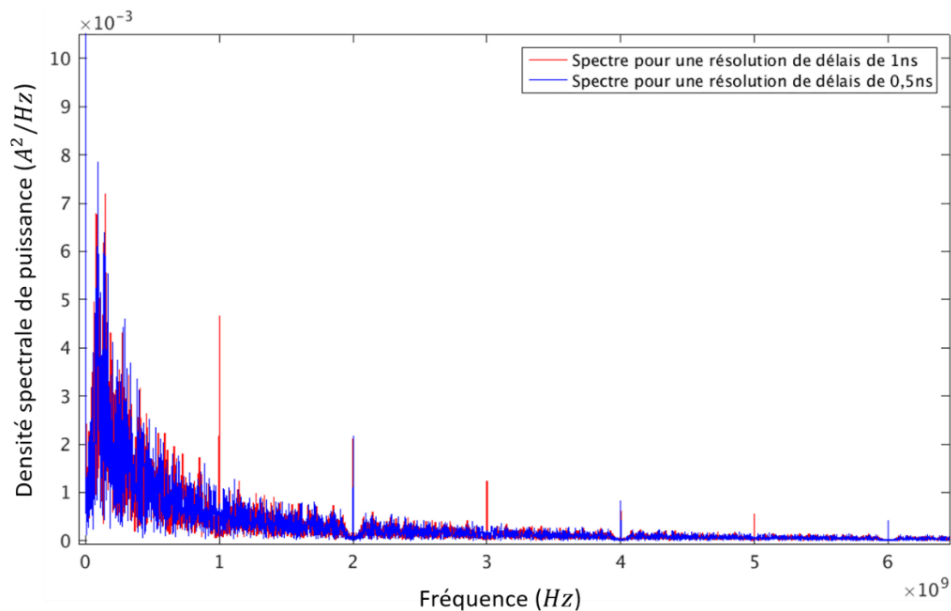


Figure V.3. Spectre pour une résolution de délais de 1 ns, en rouge, et de 0,5 ns en bleu.

De plus, pour les mêmes bornes, supérieure et inférieure, avec un pas plus petit le nombre de valeurs possibles augmente. L'algorithme génétique possède plus de combinaisons à tester, il peut trouver une solution plus facilement.

V. 2. 3. Changement d'architecture

Le concepteur peut changer l'architecture de son circuit en divisant certains étages pour aider la co-simulation à trouver une solution. Cette méthode permet de garder une consommation totale de courant constante au niveau des registres, mais en divisant certains pics et en ajoutant des délais, donc des paramètres au problème. Dans un circuit asynchrone, il est facile de diviser un étage en deux. Par exemple, le traitement d'une donnée codée sur huit bits peut se faire en deux étages au lieu d'un. Le premier étage traite les quatre premiers bits et le second les quatre autres. Cette modification est possible si les traitements des différents bits ne sont pas liés.

Pour illustrer l'impact de ce changement un circuit de référence composé de cinquante étages de huit registres a été modélisé. La co-simulation a été effectuée et une solution a été trouvée au bout de 26 générations. Ensuite, l'architecture de ce circuit a été modifiée. Cinq étages ont été divisés en deux. Le deuxième circuit possède cinquante-cinq étages, quarante-cinq avec huit registres et dix avec quatre registres. Une solution pour le même gabarit a été trouvée après 21 générations. Un dernier circuit a été évalué, où dix étages du premier circuit ont été divisés en deux. Cette fois un spectre en dessous du gabarit est trouvé après seulement 4 générations. Le Tableau V-1 récapitule ces résultats.

Nombre d'étages	Nombre de générations
50	26
55	21
60	4

Tableau V-1. Nombre de générations nécessaires pour trouver une solution en fonction du nombre d'étages dans le circuit avec une consommation de courant totale constante, de 400 registres.

En résumé, le concepteur peut adopter trois stratégies pour trouver plus facilement une combinaison de délais permettant d'obtenir un rayonnement en dessous du gabarit. Dans un premier temps, l'idéal est de concevoir le circuit en asynchrone. Cela permet d'avoir des étages combinatoires avec des chemins critiques très différents et d'obtenir une meilleure mise en forme pour le courant. Dans un second temps, le concepteur peut choisir la résolution du pas entre les valeurs des délais en fonction de son plus petit élément de délai et des pics qu'il veut éviter dans le spectre. Enfin, il peut augmenter le nombre d'étages dans son circuit en gardant le même nombre d'éléments de mémorisation. Cette stratégie permet d'augmenter les paramètres possibles pour la co-simulation et de trouver une solution plus rapidement.

V. 3. Circuit de test « Pegase »

Le premier circuit de test que nous avons réalisé est un circuit de cryptographie de norme de chiffrement avancé (*Advanced Encryption Standard, AES*). La cryptographie permet de transformer un message en clair à l'aide d'une clé en un message incompréhensible pour toute personne ne possédant pas la clé de déchiffrement. La norme de chiffrement avancé est un algorithme de chiffrement dit symétrique, il est basé sur la même clé pour crypter et décrypter le message. Un AES prend en entrée un message sur 128 bits. Nous avons implémenté un AES dit 128 bits car la clé est codée sur 128 bits. Dans d'autres cas elle peut être codée sur 192 ou 256 bits. Le message crypté en sortie est également sur 128 bits.

L'algorithme de la norme de chiffrement avancé contient quatre opérations différentes, qui sont répétées plusieurs fois [69], [70]. Le principe de son fonctionnement est représenté sur la Figure V.4. Dans un premier temps, un OU Exclusif est réalisé entre le message et la clé. Ensuite, une substitution non linéaire est réalisée pour chaque bit avec une matrice prédéfinie appelée *S-box*. Le mot de 128 bits est placé dans une matrice de quatre octets par quatre octets. La troisième étape réalise une permutation cyclique sur les lignes avec un nombre de rotations différent pour chaque ligne. Enfin, un mélange sur les colonnes est opéré en effectuant une multiplication matricielle avec une matrice prédéfinie. L'enchaînement de ces opérations est réalisé neuf fois de suite en effectuant le OU Exclusif entre la clé et le message en clair la première fois puis entre la clé et la sortie des quatre étapes. Ensuite, un dernier tour est effectué lors duquel le mélange des colonnes est remplacé par un OU Exclusif avec la clé.

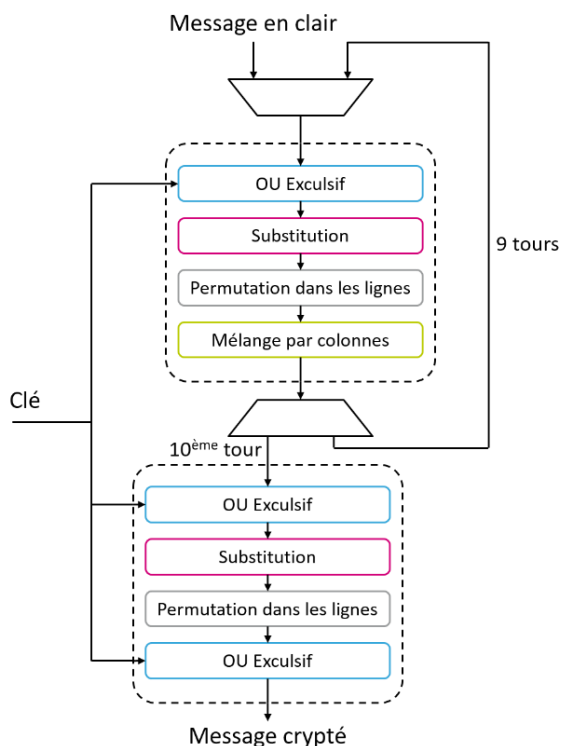


Figure V.4. Principe de fonctionnement de la norme de chiffrement avancé, toutes les flèches représentent un vecteur de 128 bits.

V. 3. 1. Architecture du circuit

Un circuit de cryptographie de norme de chiffrement avancé asynchrone a été implémenté. La seule différence dans l'architecture de notre circuit, par rapport à la Figure V.4, est le regroupement de la logique combinatoire des étages de *Substitution* et de *Permutation dans les lignes* en un seul étage. Ainsi, trois opérations sont effectuées neuf fois avant de réaliser les trois dernières opérations pour obtenir le message crypté.

Pour pouvoir effectuer neuf boucles avec les trois premières opérations, un compteur a été implémenté. De plus, l'entrée correspondant au message pour le OU Exclusif doit prendre le message en clair pour la première boucle et la sortie des trois étages pour les autres boucles. Pour réaliser cette sélection, une porte jonction avec un signal de sélection a été implémentée en entrée des trois étages. Pour choisir de reboucler ou de sortir sur les trois derniers étages, une fourche avec un signal de sélection a été implémentée entre les deux blocs de trois étages. Le signal de sélection de ces deux portes est donné par le compteur. Pour le distribuer une fourche est utilisée. La Figure V.5 représente l'architecture du réseau de contrôle de l'AES asynchrone.

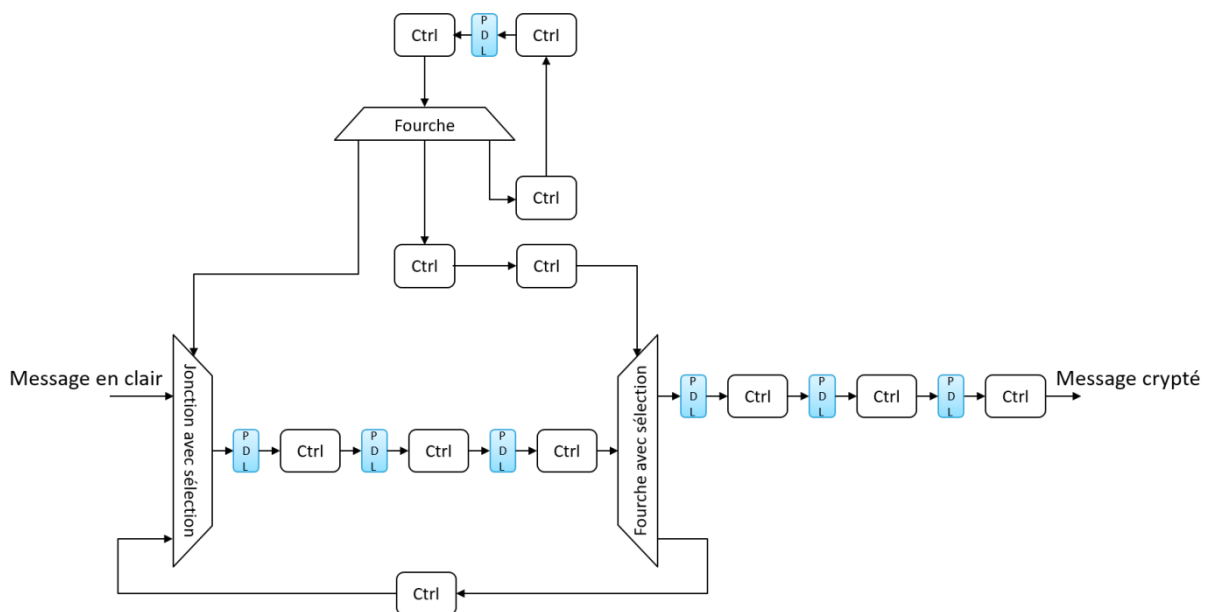


Figure V.5. Architecture de Pegase.

L'architecture présentée sur la Figure V.5 contient des blocs bleus notés « PDL ». Ils représentent des Lignes de Délais Programmables. L'objectif de ce circuit de test est de montrer que le spectre électromagnétique change lorsque les appels de courant sont à des instants différents. Les lignes de délais programmables vont nous permettre de mesurer le rayonnement du circuit pour différentes combinaisons de délais et ainsi de valider notre méthode. Nos PDL sont composées de trente-et-une portes tampons chaînées. La sortie de chaque porte tampon est également connectée sur un multiplexeur de deux vers un qui permet de choisir le retard de la requête de sortie. Celui-ci peut aller de zéro à trente-et-une fois le délai de traversé d'une porte tampon plus le délai des multiplexeurs. La Figure V.6 montre l'architecture d'une ligne de délais programmables. Pour simplifier la représentation

le multiplexeur trente-deux vers un n'a pas été décomposé en multiplexeur deux vers un. Les valeurs des codes des PDL sont stockées dans des registres. L'utilisateur peut définir les valeurs dans le code qui est exécuté par le microcontrôleur embarqué dans le circuit de test. Les valeurs des PDL présentent entre les trois premières opérations peuvent changer à chaque boucle afin d'augmenter le nombre de combinaisons possibles. Nous obtenons ainsi 31 paramètres possibles pour la co-simulation, 27 pour les trois premières opérations, 3 pour les trois dernières et 1 pour le compteur.

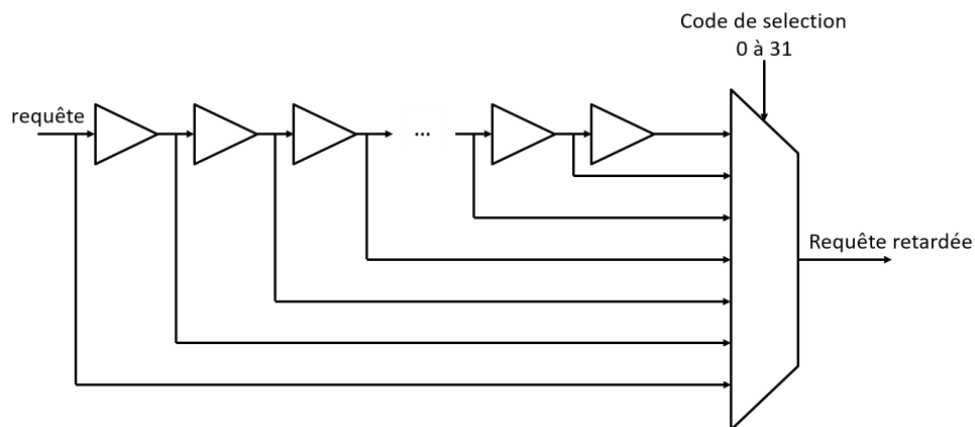


Figure V.6. Architecture possible d'une ligne de délais programmables.

Nous avons utilisé la description au niveau registres d'un circuit de norme de chiffrement avancé asynchrone déjà développé au laboratoire TIMA. Nous avons ensuite modifié l'architecture pour ajouter les lignes de délais programmables. Un signal de test a également été ajouté pour pouvoir vérifier la fonctionnalité du circuit sur notre banc de test. Il passe au niveau logique haut au début du calcul de l'AES et il redescend au niveau bas à la fin du calcul. Il permet de mesurer le temps de cryptage d'un message en fonction des valeurs des délais programmées dans le circuit.

L'AES asynchrone implémenté possède deux modes de fonctionnement. Il peut tourner en boucle. Lorsque le calcul est terminé, une requête est à nouveau envoyée automatiquement. Il peut également être lancé avec une requête externe envoyée par l'utilisateur et ensuite s'arrêter tant que l'utilisateur n'envoie pas une nouvelle requête.

V. 3. 2. Simulations

Des simulations au niveau portes ont été réalisées pour vérifier le fonctionnement de l'AES. La Figure V.7 montre la simulation d'un calcul de l'AES. Le signal rouge est notre signal de test qui sort du circuit. Il indique le temps de traitement de l'AES. Le signal en vert correspond à la requête d'entrée envoyée par l'utilisateur. Les traces bleues sont les requêtes des trois étages entre la porte de jonction et la fourche. Les neuf fronts montants pour ces requêtes correspondent aux neuf boucles du calcul. Les trois traces oranges correspondent aux trois dernières opérations du cryptage. Enfin, le signal violet correspond à la requête du compteur qui permet de reboucler neuf fois et ensuite de sortir sur les trois dernières opérations.

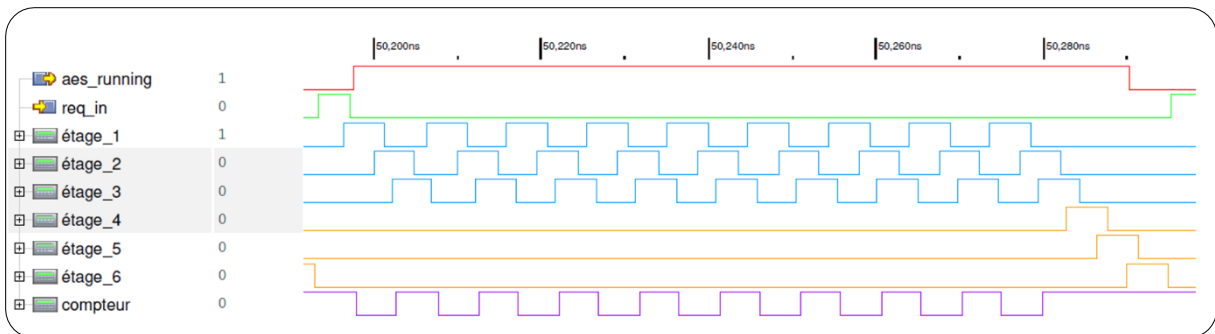


Figure V.7. Simulation du signal de test, de la requête d'entrée et des requêtes de sortie des différents étages de l'AES asynchrone.

V. 3. 3. Réalisation du circuit

Ce circuit a été réalisé au cours de ma première année de thèse, il nous a permis de mettre en place le flot de conception présenté dans le chapitre IV. Nous avons eu l'opportunité d'insérer notre AES dans un circuit de test de l'équipe. Il a été implémenté en technologie CMOS 40 nm STMicroelectronics. La Figure V.8 montre l'implémentation physique finale du circuit d'une surface de 2,89 mm². L'AES asynchrone est surligné en bleu.

Au cours de l'implémentation du circuit, nous avons rencontré des problèmes avec l'outil qui avait supprimé les portes tampons des lignes de délais programmables. Pour éviter cette optimisation, nous avons ajouté une contrainte qui interdit à l'outil d'implémentation de toucher et d'optimiser ces cellules.

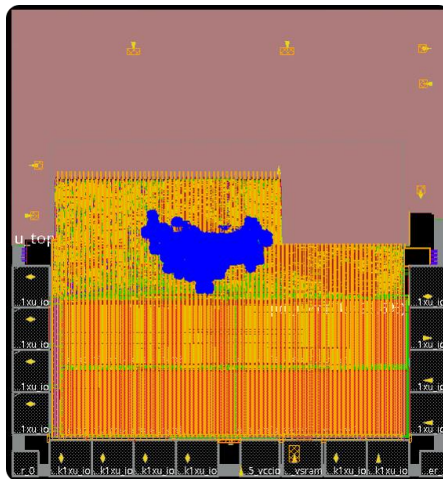


Figure V.8. Vue de l'implémentation physique du circuit, l'AES asynchrone est surligné en bleu.

L'étude de l'implémentation du circuit, nous a permis de noter des améliorations à apporter, notamment sur les contrôleurs. La Figure V.9 montre l'implémentation des contrôleurs qui activent des registres associés à la ligne de délais programmables qui se situe en amont du contrôleur. Nous pouvons constater que les cellules standards qui composent ces blocs ne sont pas toujours regroupées. Une PDL et un contrôleur n'étant pas implémentés dans une même cellule hiérarchique, l'outil ne les a pas placés côte à côte. Cela pourrait être

amélioré en ajoutant des contraintes ou en définissant un sous-bloc contenant la PDL et le contrôleur qui serait ensuite placé et routé dans le circuit.

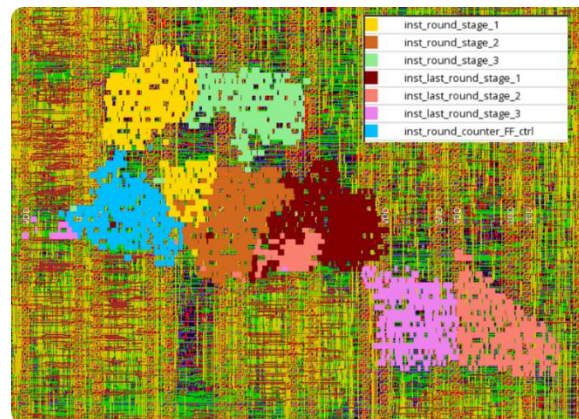


Figure V.9. Vue de l'implémentation physique des contrôleurs qui activent des registres associés à la ligne de délais programmables avant le contrôleur.

V. 4. Mesures

La Figure V.10 montre une photo du circuit fabriqué. L'AES asynchrone est entouré en rouge. A partir de cette puce nous avons pu réaliser différentes mesures. Dans un premier temps, nous avons vérifié le fonctionnement du circuit en effectuant des mesures sur notre signal de test à l'aide d'un oscilloscope. Ensuite, nous avons mesuré le courant du circuit en utilisant un port de sortie du circuit spécifiquement conçu pour mesurer le courant. Enfin, des mesures du rayonnement du spectre électromagnétique avec différents instruments de mesures et un analyseur de spectre ont été effectuées. Les sous-parties suivantes décrivent les mesures que nous avons réalisées.

Pour programmer le circuit, notamment les valeurs des codes des PDL, et pour démarrer l'AES asynchrone, nous avons utilisé le contrôleur ST-Link d'une carte Nucléo. Il nous permet de prendre la main en mode test sur le microcontrôleur du circuit et ainsi de pouvoir piloter et programmer l'AES. Une fois cette étape réalisée, la carte Nucléo peut être déconnectée sans influencer le fonctionnement du circuit. Les mémoires statiques n'étant plus utilisées par l'AES asynchrone, nous pouvons, une fois l'AES lancé, éteindre l'alimentation dédiée à ces mémoires. Ceci nous permet d'éviter un rayonnement et une consommation parasite.

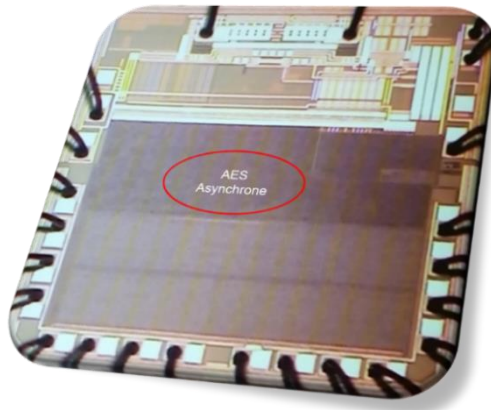


Figure V.10. Photo du circuit.

V. 4. 1. Temporelle

L'objectif des mesures temporelles était de vérifier que le circuit et les lignes de délais programmables fonctionnent. Pour cela, le délai entre le front montant, début du calcul de l'AES, et le front descendant, fin du calcul, du signal de sortie de l'AES est mesuré à l'aide d'un oscilloscope. La Figure V.11 montre la mesure du signal de test du circuit.



Figure V.11. Mesure du signal de test du circuit à l'aide d'un oscilloscope.

La Figure V.12 montre le temps de calcul de l'AES en fonction des codes des PDL. Les lignes 1, 2 et 3 correspondent aux PDL des trois premiers étages et les 4, 5 et 6 à celles des trois derniers. Comme attendu, le temps de cryptage de l'AES augmente avec les codes des PDL. De plus, nous constatons que le temps de traitement augmente plus vite lorsque les codes des PDL 1, 2 et 3 augmentent. L'algorithme de cryptographie passe neuf fois dans les trois premiers étages et une fois seulement dans les trois derniers. Le délai ajouté entre la jonction et la fourche est donc cumulé neuf fois. C'est pourquoi les trois premières PDL ont un impact plus fort sur le temps de calcul.

La ligne de délais programmables insérée dans le compteur n'est pas représentée dans cette mesure, sa valeur est constante. Le changement de son code n'a pas d'impact sur le

calcul du message car le délai dans le compteur est inférieur au délai des trois premiers étages. La jonction et la fourche ne se trouvent jamais en attente du signal de sélection. De plus, le contrôleur du compteur active seulement quatre registres, il n'a donc pas une grande contribution au spectre électromagnétique. Lors des mesures nous laisserons donc cette PDL à zéro.

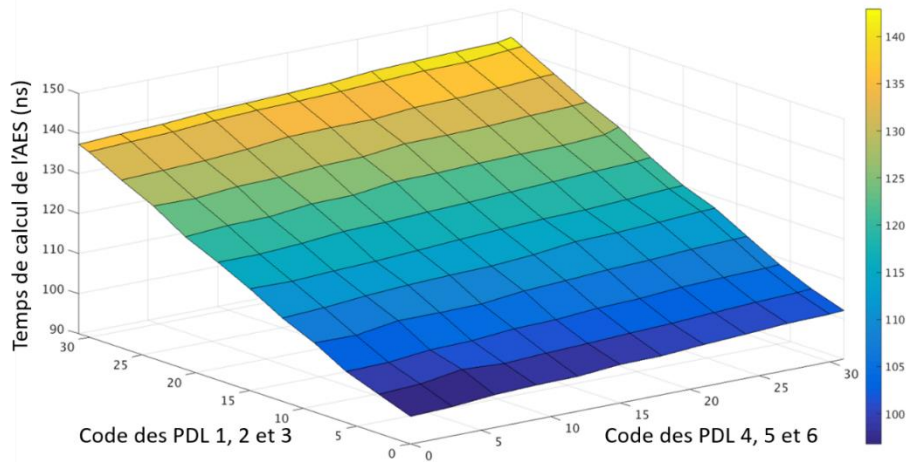


Figure V.12. Mesure du temps de calcul de l'AES en fonction des codes des lignes de délais programmables.

Une fois le fonctionnement du circuit validé, avec la possibilité de faire varier les valeurs des délais, donc de déplacer les appels de courant, nous avons effectué des mesures de courant du circuit.

V. 4. 2. Courant

Un port du circuit de test a été prévu pour mesurer une image du courant au niveau du régulateur à faible chute de tension avec un ratio de cent dû au nombre de transistors utilisés pour le miroir de courant. Une résistance de 11 kΩ a été soudée à la sortie de ce port. Nous avons mesuré la tension à ses bornes au moment du démarrage du circuit, avant que l'AES démarre et pendant le calcul de l'AES, Figure V.13. Ainsi nous pouvons calculer le courant moyen consommé grâce à l'Équation V-5 où I_{total} est le courant pendant le calcul de l'AES et $I_{statique}$ le courant consommé par le circuit dès le démarrage, sans l'AES.

$$I_{AES} = I_{total} - I_{statique} \quad \text{Équation V-5}$$

Les courants sont obtenus en utilisant l'Équation V-6 où Res correspond à la résistance de 11 kΩ et $Ratio$ est le coefficient dû à la structure utilisée pour le port de sortie.

$$I_n = \frac{U_n}{Res} * Ratio \quad \text{Équation V-6}$$

La tension au moment du démarrage du circuit est de 78 mV, mesurée sur la Figure V.13.a. La tension lors du calcul de l'AES est de 368 mV, Figure V.13.b. L'AES consomme alors un courant de 2,636 mA.

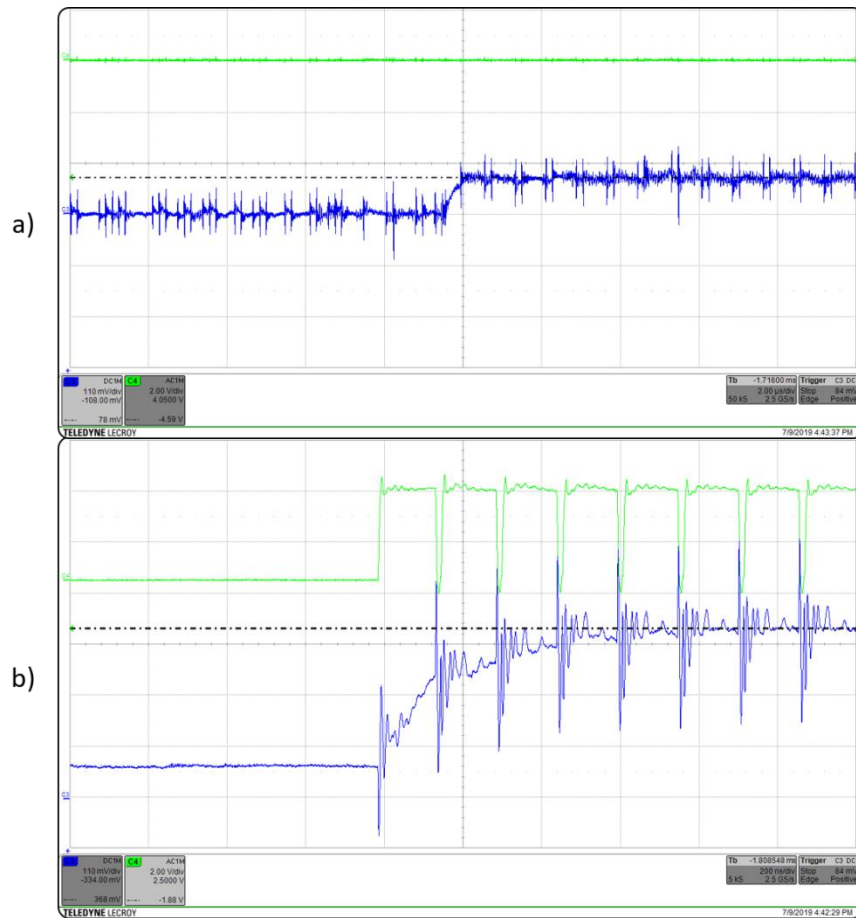


Figure V.13. Mesure du courant en bleu et du signal de test en vert au a) démarrage du circuit b) démarrage de l'AES asynchrone.

Nous avons ensuite démarré la capture de l'oscilloscope sur le signal de test de l'AES pour obtenir plus précisément le courant d'un seul calcul. La Figure V.14 montre un calcul de l'AES, pour un code de PDL de 0x1F, avec le courant associé. Nous ne pouvons pas identifier la consommation des différents étages à l'aide du courant. Le problème vient du fait que le circuit possède un régulateur à faible chute de tension (*low-dropout regulator, LDO*) ainsi qu'une capacité de 3 nF qui lissent la consommation de courant.

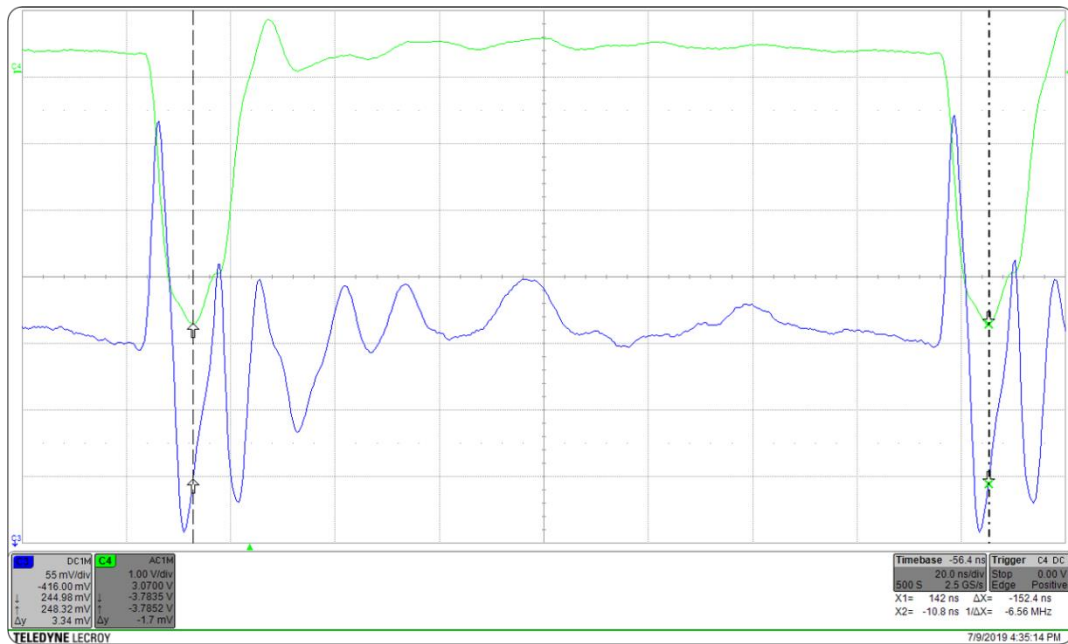


Figure V.14. Mesure du signal de test en vert et du courant associé en bleu pour les PDL à 0x1F.

Les mesures en courant nous ont permis de repérer un motif qui est reproduit à chaque calcul de l’AES. En revanche, à cause du régulateur à faible chute de tension et de la capacité de 3 nF, nous n’avons pas pu valider notre modèle de pic de courant.

Nous avons ensuite réalisé des mesures électromagnétiques, avec différents instruments de mesure, pour valider notre méthode de conception.

V. 4. 3. Electromagnétique

Les premiers tests électromagnétiques du circuit ont été réalisés dans une cage de Faraday, avec le circuit câblé sur une plaque à trou. Ces tests nous ont permis de valider le fonctionnement du circuit avec une mise en place simple. Cependant, ils ne nous ont pas permis de mesurer de façon correcte le rayonnement. La cage de Faraday était mal isolée et l’antenne captait un rayonnement même avec le circuit éteint. De plus, une plaque à trou n’est pas bien adaptée à des mesures électromagnétiques.

Nous avons alors décidé de concevoir une carte de test spécifique à notre circuit (Figure V.15). Pour limiter les rayonnements parasites nous avons fait en sorte d’avoir toutes les pistes de connexions du même côté et d’avoir un plan de masse de l’autre. Nous avons la possibilité de souder le circuit du côté du plan de masse. Nous verrons par la suite l’utilité de cette contrainte pour les mesures avec la cellule de mesure électromagnétique transverse (*Transverse ElectroMagnetic, TEM*).

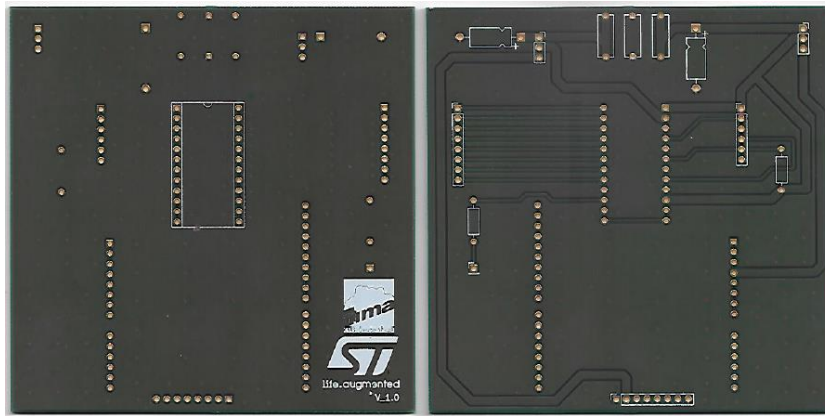


Figure V.15. Photos de notre carte de test, a) face avec le plan de masse où le circuit sera soudé b) face où les pistes sont routées et où les composants seront soudés.

V. 4. 3. 1. Mini cage de Faraday

Pour mieux isoler le circuit et éviter de mesurer le rayonnement de l'environnement, nous avons utilisé une mini-cage de Faraday. Une antenne, une boucle simple réalisée avec un fil, est placée au-dessus du circuit de test comme montré sur la Figure V.16. Elle est ensuite reliée à un analyseur de spectre.

Néanmoins, l'antenne n'étant pas fixée dans la mini cage, sa position variait entre les différents tests. Ainsi, lors des reprogrammations des combinaisons de délais du circuit, l'antenne ne se repositionnait pas exactement de la même manière au-dessus du circuit. Ce problème rendait les résultats difficilement comparables et les tests non reproductibles. De plus, l'antenne n'était pas caractérisée. De ce fait, nous ne pouvions pas savoir si elle était sensible aux bonnes plages de fréquences. Ces inconvénients ont fait que cette méthode n'a pas donné de résultats exploitables. Elle a néanmoins permis de visualiser (de façon qualitative) un déplacement de spectre en fonction des codes des PDL.

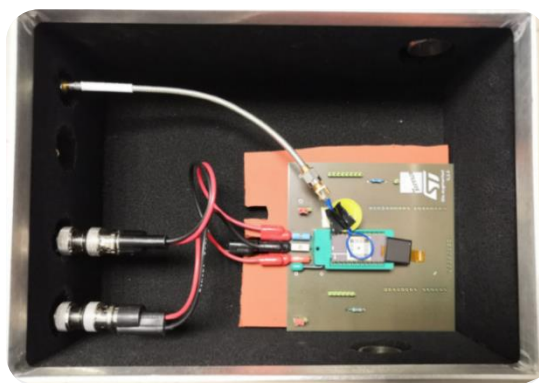


Figure V.16. Test dans une mini cage de Faraday avec une antenne reliée à un analyseur de spectre.

V. 4. 3. 2. Cellule TEM

Le deuxième test a été réalisé avec une cellule de mesure ElectroMagnétique Transverse reliée à un analyseur de spectre (Figure V.17). Notre carte de test permet d'avoir seulement le circuit à l'intérieur de la cellule et les autres composants en dehors. Ainsi, l'AES peut être reprogrammé facilement et le rayonnement des pistes de la carte n'est pas mesuré dans la cellule.

Cependant, nous avons également été confrontés à des problèmes durant ces mesures. La carte de test que nous avons conçue est vernie donc le contact avec la masse de la cellule ne se faisait pas, ce qui ne permet pas de bien isoler la cellule des rayonnements de l'environnement. Nous avons donc enlevé le vernis sur le contour de la carte pour faire le contact.

De plus, les contacts pour les alimentations du circuit sont sur le bord de la carte, ce qui crée un court-circuit avec la masse de la cellule TEM. Pour éviter ce problème nous avons isolé les contacts avec du chatterton. Par conséquent, nous ne pouvions pas avoir un contact avec la masse de la cellule TEM sur tout le contour de la carte. Le rayonnement de notre circuit étant faible, il est donc très difficile à mesurer. Les mesures avec la cellule TEM n'ont donc pas donné non plus de résultats exploitables.

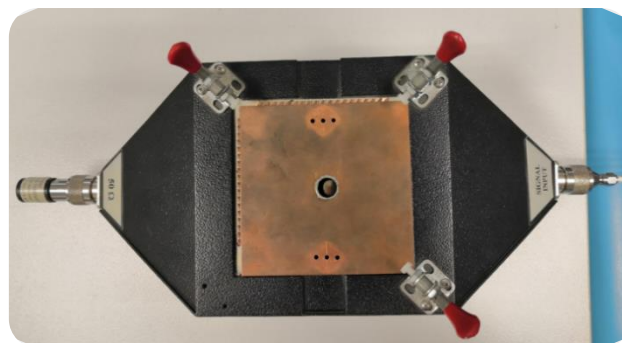
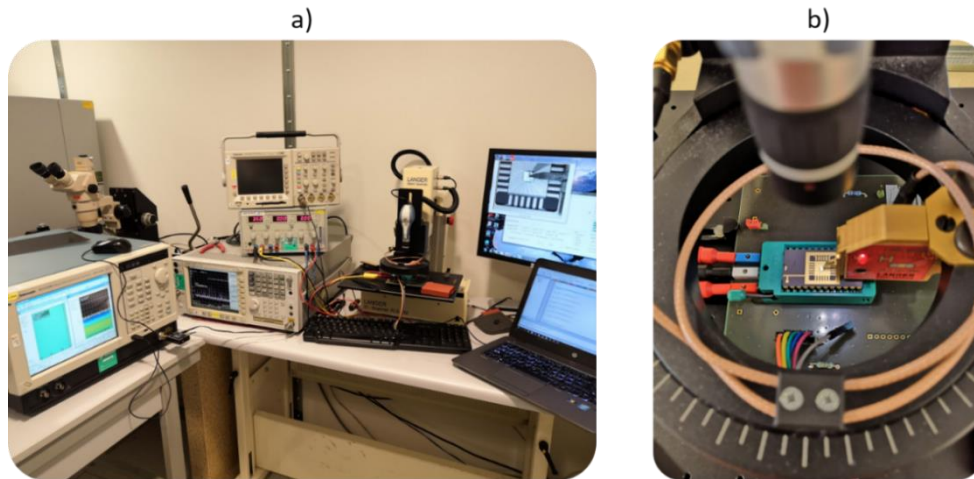


Figure V.17. Cellule de mesure Electromagnétique Transverse.

V. 4. 3. 3. Scanner de surface

Nous avons fait des mesures avec un scanner de surface, comme le montre la Figure V.18. Une pointe avec une boucle de 150 μm de diamètre se déplace au-dessus du circuit avec les paramètres définis par l'utilisateur. Une caméra se trouve au-dessus de la pointe pour pouvoir la placer précisément au-dessus de la partie à mesurer.



**Figure V.18. a) Vue du banc de test avec les analyseurs de spectre et le scanner de surface
b) vue de dessus de la pointe du scanner de surface avec le circuit.**

Ce test nous a permis de nous placer précisément au-dessus de l'AES pour effectuer les mesures. La proximité de la pointe avec le circuit nous a permis d'avoir de meilleures mesures spectrales.

Dans un premier temps, nous avons réalisé des mesures avec un analyseur de spectre temps réel relié au scanner de surface. Cet appareil de mesure nous a permis de voir l'évolution des fréquences dans le temps lorsqu'on change les combinaisons de délais. Le spectrogramme permet de visualiser les fréquences en abscisse et le temps en ordonnée.

La Figure V.19.a montre l'évolution des fréquences rayonnées dans le temps. La première zone en haut correspond au rayonnement lorsque toutes les PDL ont un code de 0x0F. Ensuite, nous avons reprogrammé le circuit, deuxième zone encadrée en pointillés, pour mesurer la solution trouvée par la co-simulation pour la plage de fréquences 230-280 MHz. Le rayonnement a diminué grâce à cette combinaison. Seul un pic reste élevé. Or, ce pic est toujours présent dans la dernière zone, qui correspond au rayonnement du circuit lorsque l'AES asynchrone n'est pas activé. Nous pouvons en déduire que cette fréquence n'est pas rayonnée par notre AES mais par une autre partie du circuit qui ne correspond pas à nos travaux. La Figure V.19.b montre le spectre pour chaque combinaison.

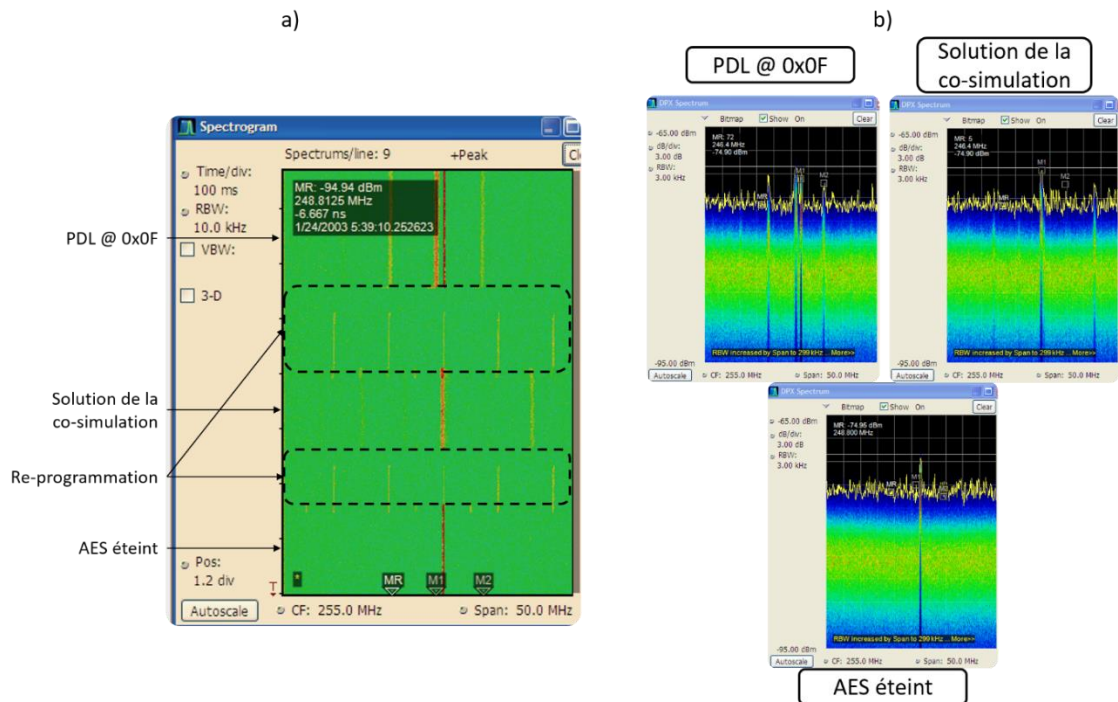


Figure V.19. Mesures avec l'analyseur de spectre temps réel sur la plage de fréquences 370-400 MHz.

Dans un deuxième temps, nous avons réalisé des mesures avec un analyseur de spectre séries (*PSA series spectrum analyzer*) relié au scanner de surface. Pour obtenir des mesures précises, nous avons utilisé une résolution en fréquence de 10 kHz. Cela augmente le temps de balayage de l'analyseur qui devient lent par rapport au temps de calcul de l'AES. Pour mesurer le rayonnement et ne pas manquer de pic, à cause de la différence temporelle entre le balayage et le calcul, nous avons utilisé la fonction « *max hold* » de l'analyseur de spectre. Elle nous permet de garder à chaque balayage le point maximum pour chaque fréquence, comme le montre la Figure V.20. Celle-ci montre le rayonnement du circuit sur la plage de fréquences 1-25 MHz, pour trois combinaisons de délais différentes. Pour chaque trace, les codes de toutes les lignes de délais programmables sont identiques, pour la trace bleue à 0x1F donc à 32 en décimale, pour la trace jaune à 0x17 donc 23 et la rose à 0x0F qui correspond à 15 portes tampons. Nous constatons que plus le calcul est lent, avec un code élevé, plus le spectre se déplace vers les basses fréquences. L'amplitude du pic augmente lorsque le nombre de portes tampons activées augmente, le circuit consomme plus.

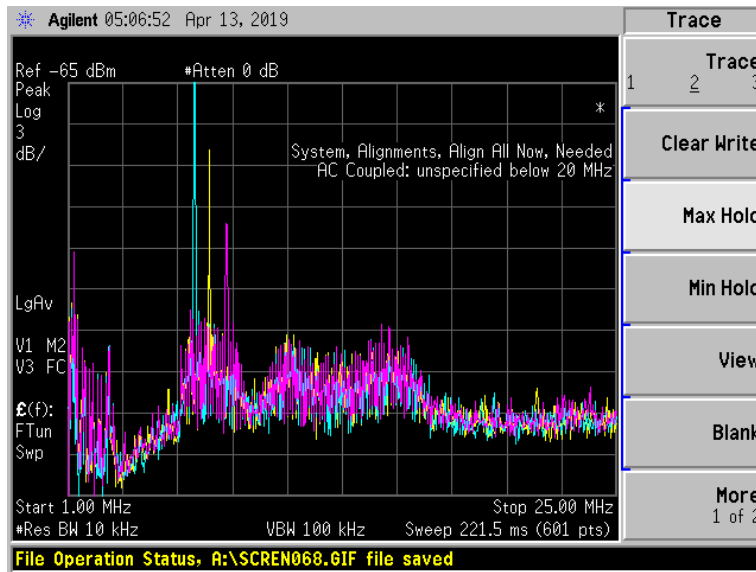


Figure V.20. Capture d'écran de l'analyseur de spectre sur 1-25 MHz, les PDL sont à 0x1F pour la trace bleue, 0x17 pour la jaune et 0x0F pour la rose.

Pour chaque combinaison, ce pic basse fréquence correspond à la fréquence de calcul de l'AES. Nous l'avons confirmé en mesurant le signal de test de chaque combinaison grâce à un oscilloscope, Figure V.21. Le Tableau V-2 récapitule les fréquences mesurées avec l'analyseur de spectre, la période correspondante et la période mesurée sur le signal de test. Nous constatons pour le code 0x0F que les valeurs sont similaires à 0,2 ns près dû à la précision des mesures.

Code des PDL (hexadécimale)	Fréquence mesurée (MHz)	Calcul de la période (ns)	Période mesurée à l'oscilloscope (ns)
0x0F	7,92	126,2	126
0x17	7,16	139,6	139,6
0x1F	6,56	152,4	152,4

Tableau V-2. Récapitulatif des fréquences mesurées grâce au scanner de surface comparées aux périodes mesurées à l'oscilloscope.

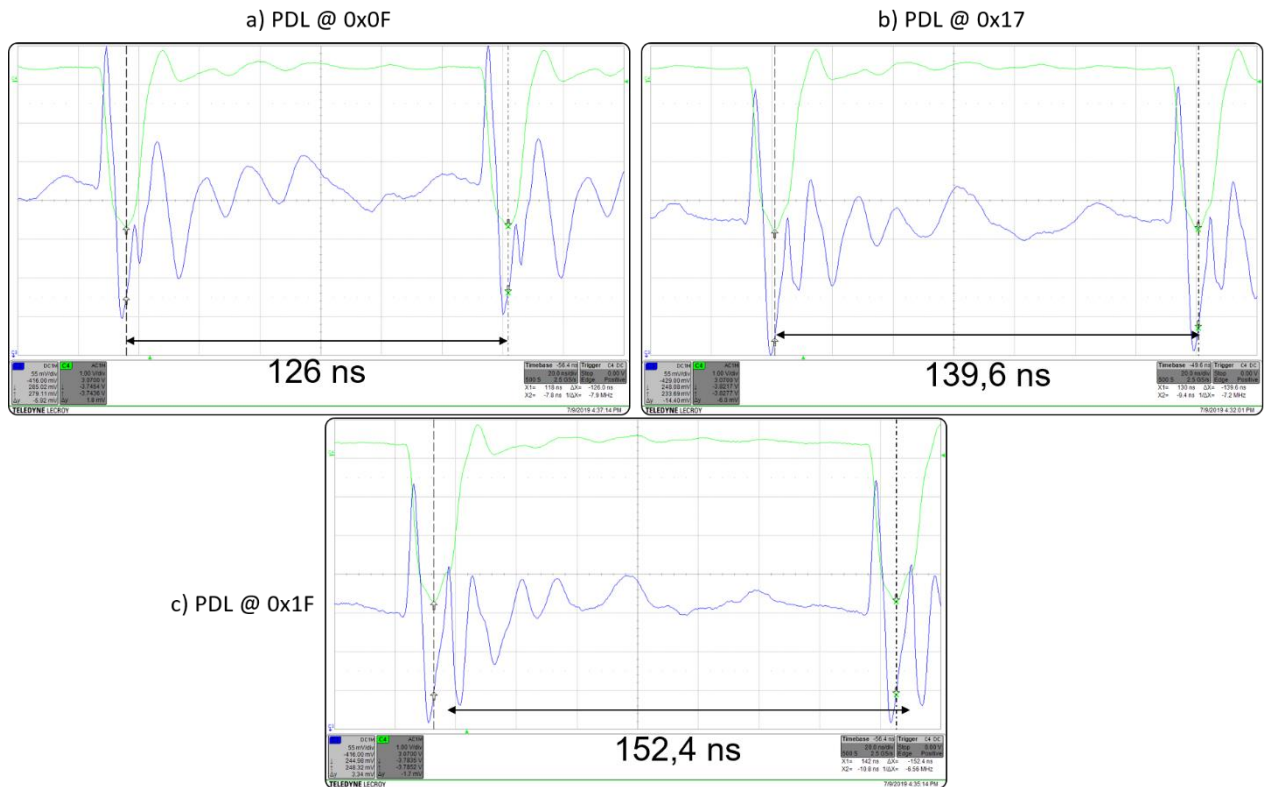


Figure V.21. Mesures du signal de test pour des codes de PDL à a) 0x0F b) 0x17 c) 0x1F.

L'analyseur de spectre permet de récupérer les informations de chaque point de la courbe sous forme de fichier CSV (Valeurs Séparées par des Virgules). Le traitement de ces fichiers nous a permis de soustraire le rayonnement du circuit lorsque l'AES asynchrone n'est pas en fonctionnement, au rayonnement lorsqu'il est activé. La Figure V.22.a montre le rayonnement du circuit avec quatre combinaisons de délais différentes ainsi que le rayonnement lorsque l'AES est éteint, courbe verte. La Figure V.22.b correspond au rayonnement du circuit après la soustraction du spectre lorsque le circuit est éteint.

La soustraction entre les mesures avec et sans le fonctionnement de l'AES, est effectuée pour les représentations des mesures suivantes. Elle permet d'avoir le rayonnement réel de notre circuit, sans le rayonnement parasite du reste du circuit de test.

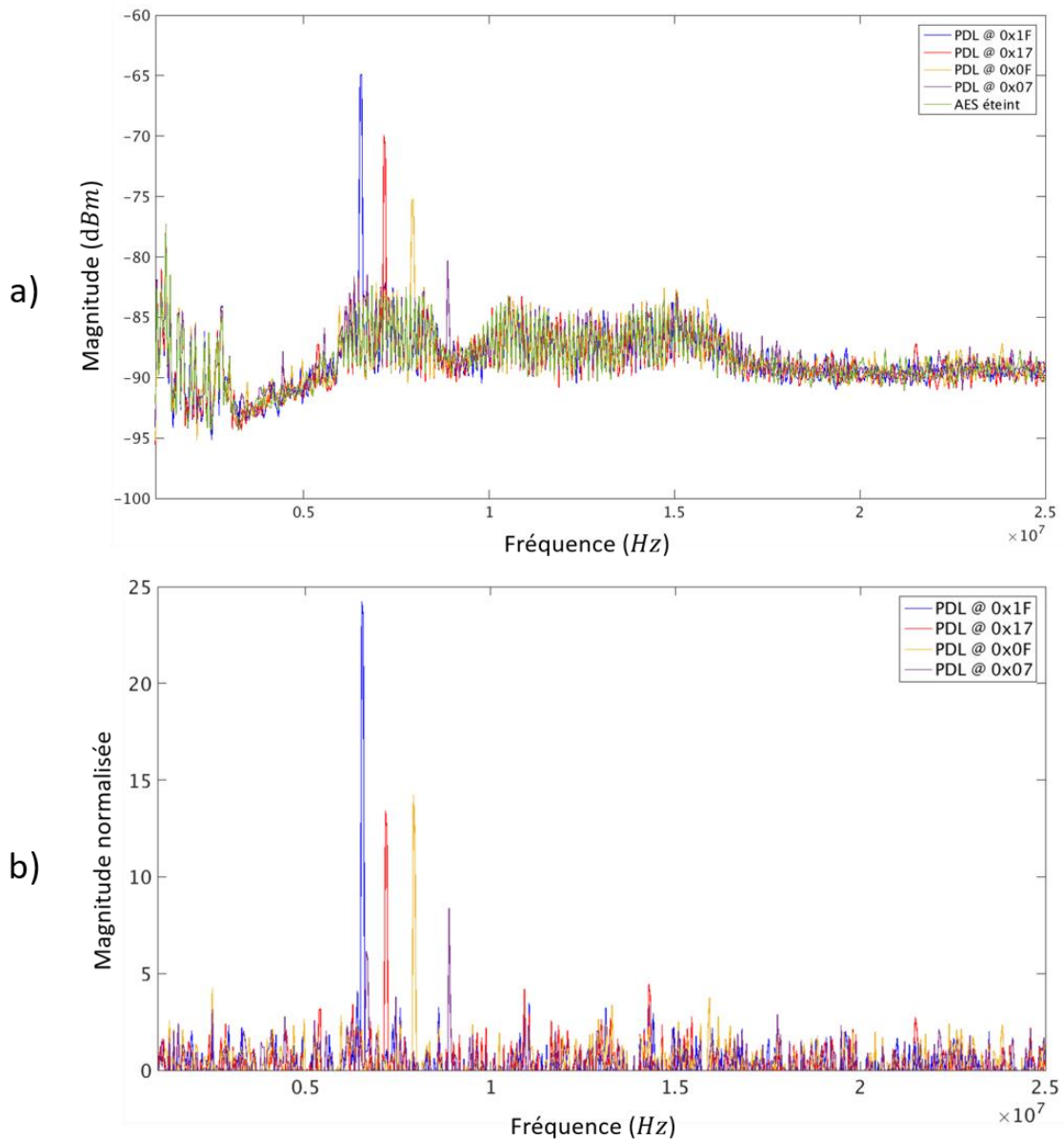


Figure V.22. a) Mesures du spectre électromagnétique du circuit entre 1 MHz et 25 MHz et b) soustraction du spectre sans l’AES pour obtenir seulement le rayonnement de l’AES.

Ensuite, nous avons utilisé différents gabarits en fréquences pour réduire le rayonnement sur certains intervalles. L’étape de co-simulation nous a permis de trouver les codes des PDL à utiliser. Nous avons effectué cette mise en forme du spectre sur les plages 210-230 MHz, 230-280 MHz et 370-400 MHz. La Figure V.23 montre les résultats obtenus au cours des mesures de ces combinaisons. Pour chaque courbe la trace bleue correspond à la combinaison de délais trouvée au cours de la co-simulation. Pour certaines combinaisons nous avons rajouté des portes tampons dans les délais programmables pour pallier aux variations de fabrications et décaler des pics qui se trouvaient dans la zone à diminuer. Nous avons comparé ces courbes avec des combinaisons qui rayonnent dans les plages de fréquences visées. Ces mesures montrent que notre méthode permet de mettre en forme le spectre en décalant les appels de courant.

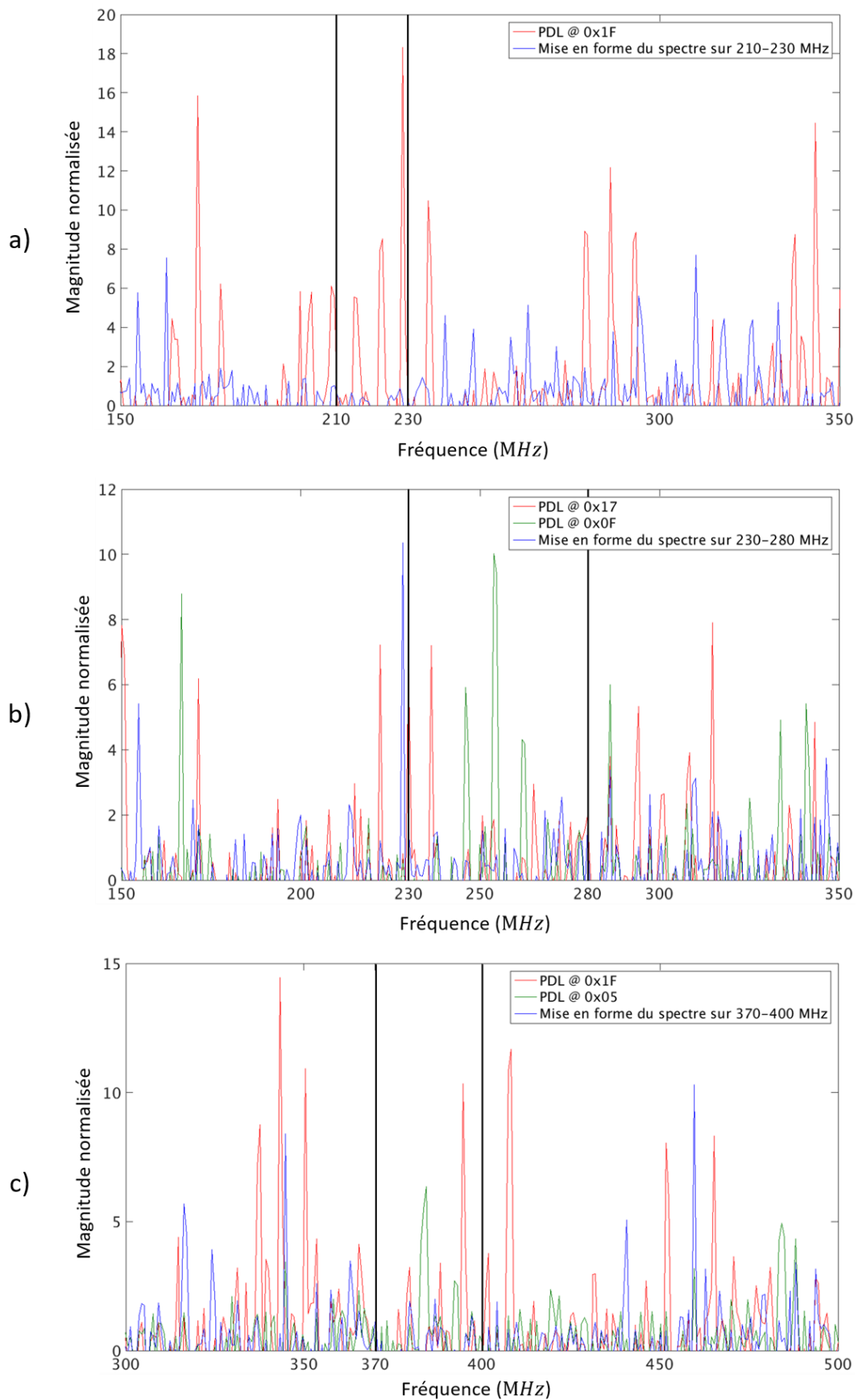


Figure V.23. Mise en forme du spectre grâce à la co-simulation sur les plages de fréquences a) 210-230 MHz, b) 230-280 MHz et c) 370-400 MHz.

Enfin, nous avons mesuré le rayonnement en fixant les codes de toutes les PDL à 0x07 et en changeant le texte à crypter. Cette mesure nous permet de voir l'impact du nombre de registres qui commutent sur le spectre. La Figure V.24 montre le rayonnement pour trois textes différents. Nous constatons que l'amplitude des pics varie mais qu'il n'y a pas de nouveau pic avec un fort rayonnement.

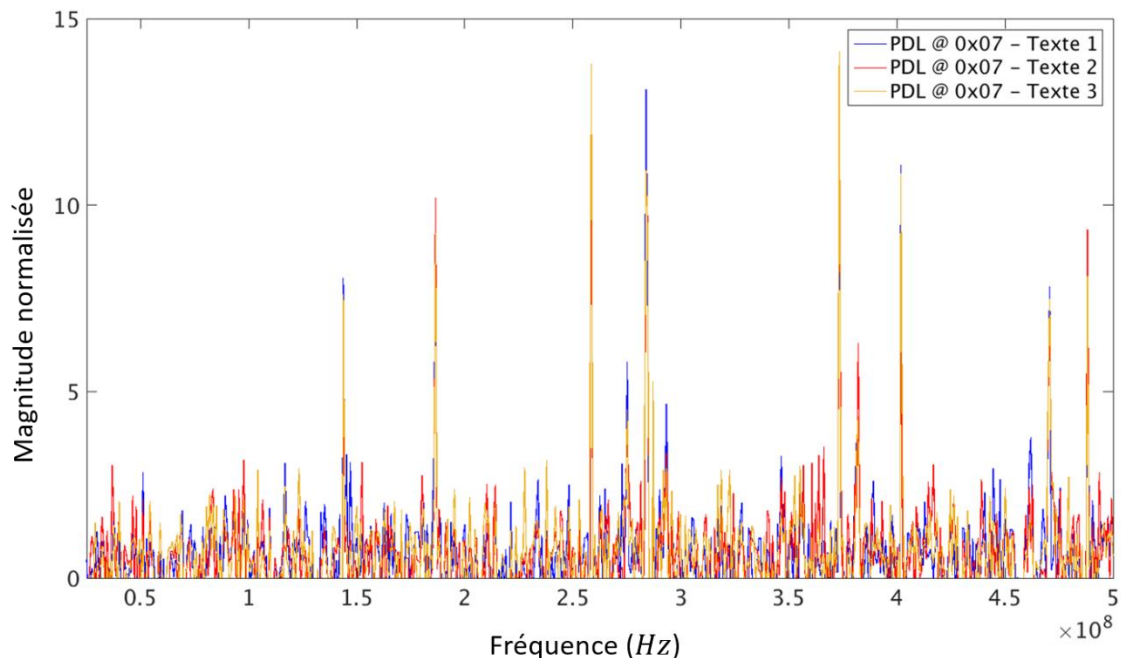


Figure V.24. Mesure du rayonnement électromagnétique avec un code de 0x07 pour les PDL et trois messages différents à crypter.

Nous avons montré grâce à trois configurations de délais, obtenues à l'aide de la co-simulation, que nous arrivons à mettre en forme le spectre électromagnétique pour diminuer le rayonnement sur des plages de fréquences spécifiques. Ces travaux sur la modification du rayonnement électromagnétique ont donné lieu à un dépôt de brevet qui est en cours.

V. 5. Circuit de test « Hydra »

Suite aux difficultés rencontrées au cours des mesures de courant du premier circuit de test, nous avons décidé de concevoir un nouveau circuit plus adapté. L'objectif de ce circuit est d'arriver à corréliser plus facilement les mesures de courant et électromagnétiques avec les simulations, et d'explorer différentes conditions de fonctionnement du circuit (dues aux variations de procédés pendant la fabrication). Pour cela, nous avons décidé d'implémenter un circuit plus simple, linéaire, et avec de nombreux registres qui commutent pour produire de forts appels de courant et un rayonnement plus important.

V. 5. 1. Architecture du circuit

Le chemin de données de ce circuit est très simple, un OU Exclusif prend en entrée un bit de polarité défini à l'initialisation du circuit et la sortie de l'étage précédent, la sortie de la porte est mémorisée. Cinq étages sont réalisés de cette manière et le dernier reboucle sur le premier. Les valeurs à l'initialisation des registres valent 0. Le bit de polarité permet de choisir quels étages de registres vont commuter ou non. Le chemin de données est représenté en noir sur la Figure V.25.

Le chemin de contrôle est composé de cinq portes de Muller, chacune associée à une ligne de délais programmables, comme montré en bleu sur la Figure V.25. Le circuit ne possède pas de port d'entrée de requête. Une partie combinatoire est ajoutée avant la première porte de Muller pour générer la requête d'entrée de celle-ci et assurer le respect du protocole. Elle est générée en entrée lorsque la requête de sortie de la dernière porte de Muller est au niveau logique haut pour s'assurer que le dernier étage a eu le temps d'effectuer son calcul. Une deuxième partie combinatoire est insérée à la fin du chemin de contrôle pour générer l'acquiescement du dernier contrôleur. Tout comme pour le premier circuit de test, les lignes de délais programmables servent à tester plusieurs combinaisons de délais lors des mesures.

Pour définir les valeurs pendant l'initialisation, un registre à décalage est conçu. Il est représenté en rose sur la Figure V.25. Une horloge est utilisée seulement pour faire passer les valeurs d'un registre à l'autre. Une fois que toutes les valeurs sont chargées dans le registre à décalage, l'horloge est coupée pour ne pas créer une consommation et un rayonnement parasites. La sortie de chaque registre est reliée à une porte du circuit pour définir sa valeur d'initialisation. Cette méthode permet de réduire le nombre de ports d'entrée nécessaires pour faire fonctionner le circuit. Finalement, nous avons besoin de quatre ports d'entrée. Deux pour les signaux d'initialisation, un pour la partie de donnée et l'autre pour la partie contrôle. Pour une question de lisibilité ces ports ne sont pas reliés sur la Figure V.25. Nous devons ajouter un port pour l'horloge du registre à décalage et un autre pour l'entrée du premier registre. Nous avons également besoin de deux ports de sortie. Un premier pour la requête du cinquième contrôleur qui nous permet de vérifier que le circuit fonctionne correctement. Cette requête est également redirigée vers la partie combinatoire d'entrée pour relancer une requête en entrée. Le deuxième port de sortie nous permet de récupérer la valeur du dernier registre du circuit d'initialisation. Cette valeur nous permet de savoir que tous les registres sont bien initialisés avec les valeurs du circuit.

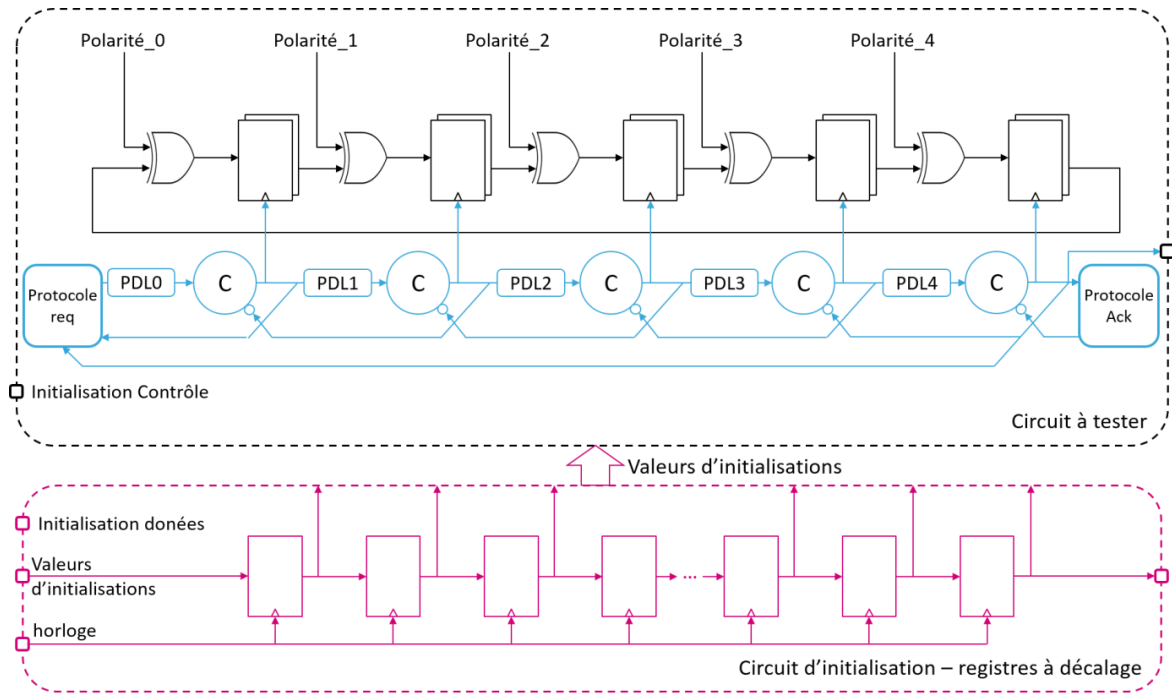


Figure V.25. Architecture du deuxième circuit de test *Hydra*.

V. 5. 2. Simulation

Des simulations ont été réalisées pour valider le fonctionnement du circuit. Les valeurs des PDL présentées dans le Tableau V-3 sont utilisées dans les simulations présentées sur la Figure V.26. La partie *a)* de la figure montre la modélisation du courant à l'étape de co-simulation et la partie *b)* la simulation au niveau portes du circuit.

PDL	Valeur (décimale)
0	14
1	24
2	6
3	0
4	7

Tableau V-3. Valeurs des PDL des simulations de la Figure V.26.

Nous constatons que les écarts entre les pics de courant consécutifs, pour la modélisation, et les fronts montants des signaux qui activent les registres, pour la simulation, sont les mêmes à quelques picosecondes près. La différence entre les valeurs est due au pas d'échantillonnage temporel utilisé pour modéliser le courant. Ce pas ne permet pas toujours de récupérer le point le plus haut du pic de courant. Cette valeur a été choisie pour avoir une bonne résolution tout en ayant des temps de calcul raisonnables. En revanche, nous remarquons des différences sur les instants des fronts descendants, une amélioration sur la récupération des délais des signaux d'acquittement est nécessaire. Le rayonnement électromagnétique étant

principalement dû aux fronts montants, nous avons, dans un premier temps, centré notre étude sur les fronts montants et non sur les descendants.

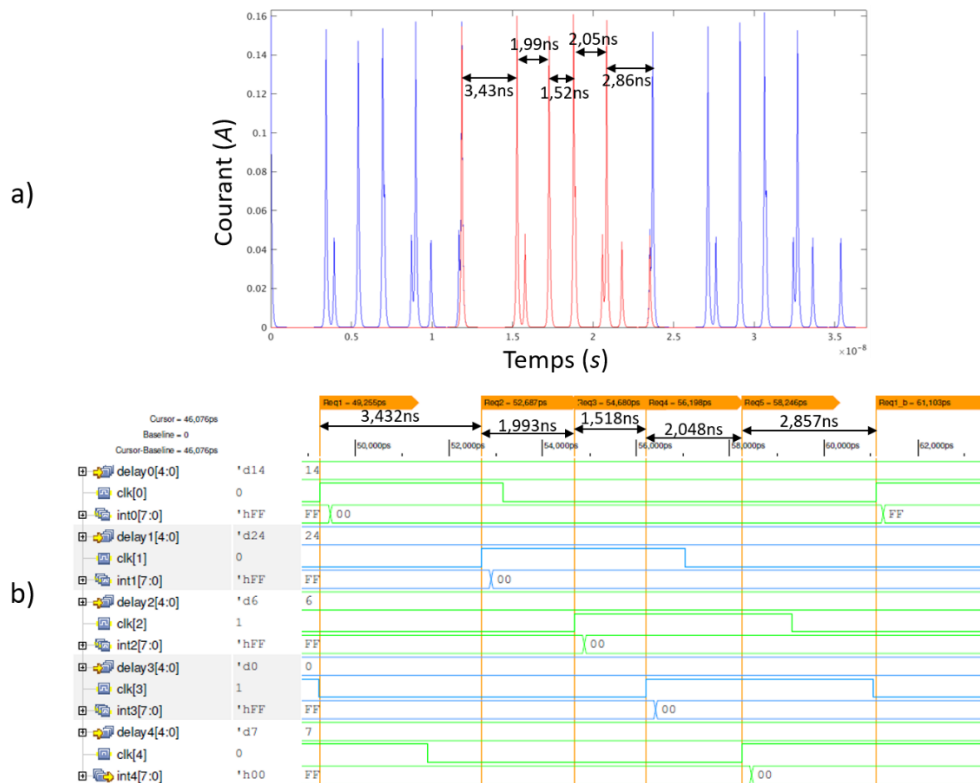


Figure V.26. a) Modélisation du courant et b) Simulation logique du circuit au niveau portes avec les mêmes délais dans les PDL.

V. 5. 3. Réalisation du circuit

Le circuit a été réalisé avec le flot de conception présenté dans le chapitre IV. L'étape de co-simulation nous permet de trouver différentes combinaisons à tester pour obtenir des spectres différents afin de cibler des plages de rayonnement spécifiques. Le circuit étant petit et répétitif, l'étape d'implémentation physique a été réalisée à la main. De cette manière, les portes standards qui composent les contrôleurs et les PDL ont été regroupées. Le circuit de test a été implémenté dans deux technologies différentes, en BiCMOS 55 nm STMicroelectronics et en CMOS 40 nm STMicroelectronics. Nous avons ciblé deux opportunités pour fabriquer notre circuit. Malheureusement, aucune des deux ne s'est concrétisée.

La version en 55 nm a été conçue avec 512 registres par étage. Chaque étage possède 4 bits de polarité qui vont chacun vers 128 OU Exclusif. Cela permet de faire commuter 128, 256, 384 ou 512 registres par étage. La Figure V.27 montre l'implémentation finale du circuit en 55 nm. Chaque rectangle noir possède 128 fois les cinq étages de registres et de OU Exclusifs. Les rectangles roses montrent les cinq contrôleurs et l'ovale rouge indique les registres à décalages. La surface du circuit est de $70 \mu\text{m}^2$.

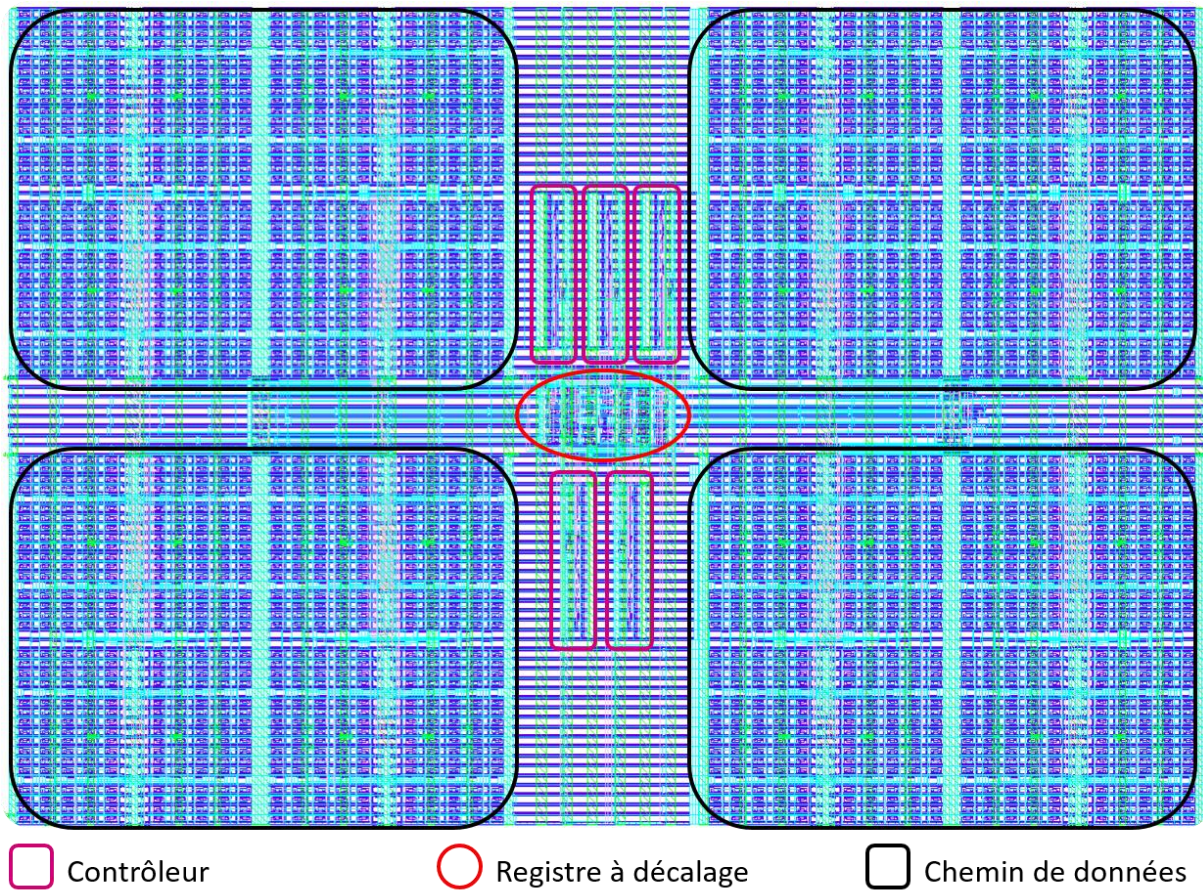


Figure V.27. Implémentation physique du circuit en BiCMOS 55 nm STMicroelectronics.

Une version plus petite a été implémentée en 40 nm pour pouvoir faire des simulations après l'implémentation physique assez rapidement. Chaque étage possède seulement 8 registres et un bit de polarité. La Figure V.28 montre l'implémentation regroupée d'une ligne de délais programmables et du contrôleur suivant. La réalisation d'un sous-bloc reproduit cinq fois à l'identique permet de diminuer les différences de propagations entre les contrôleurs.

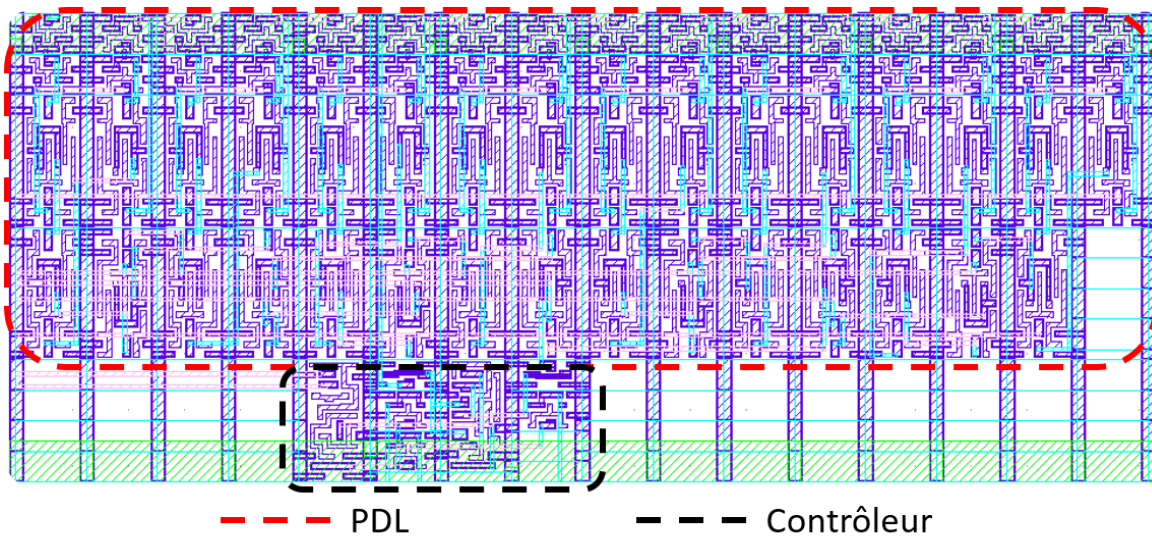


Figure V.28. Implémentation d'une ligne de délais programmables et d'un contrôleur en CMOS 40 nm STMicroelectronics.

Les deux versions du circuit sont prêtes à être envoyées en fabrication. La réalisation de ce circuit de test nous permettrait de réaliser de nouvelles mesures et de valider notre modèle de courant. Il serait intégré dans un circuit avec une alimentation qui lui serait dédiée pour éviter d'avoir à nouveau les effets indésirables du régulateur à faible chute de tension pour nos mesures et pour être mieux isolé du reste du circuit.

V. 5. 4. Simulation après implémentation

Nous avons effectué une extraction des parasites après placement et routage. Cette extraction nous fournit les capacités entre les fils dans les différentes conditions de fonctionnement du circuit (meilleur, typique et pire cas). Cette extraction nous a permis de réaliser des simulations pour extraire la consommation de courant et comparer les spectres électromagnétiques.

Nous avons simulé le circuit dans le pire cas de fonctionnement sans ajouter de délais dans les PDL. La Figure V.29.a représente la requête de sortie du premier étage en vert et du deuxième en jaune. Ensuite, des simulations dans le meilleur cas ont été effectuées en ajoutant des délais à l'aide des PDL, le même code est appliqué pour toutes les lignes de délais. L'objectif en ajoutant du délai est de retrouver l'écart entre les fronts des requêtes, donc entre les appels de courant, pour obtenir le spectre désiré. En ajoutant une porte tampon avec les PDL, nous obtenons un écart de 1,24 ns (Figure V.29.b) alors que dans le pire cas nous avons un délai de 1,22 ns. Le spectre dans ces conditions, en bleu sur la Figure V.29.c, est comparé avec le spectre dans le pire cas de fonctionnement, en vert. Nous constatons qu'ils sont très proches au niveau des fréquences rayonnées.

En optimisant les lignes de délais programmables, il est possible de rajouter du délai, si le circuit est plus rapide que prévu, pour se ramener aux délais calculés lors de la co-simulation.

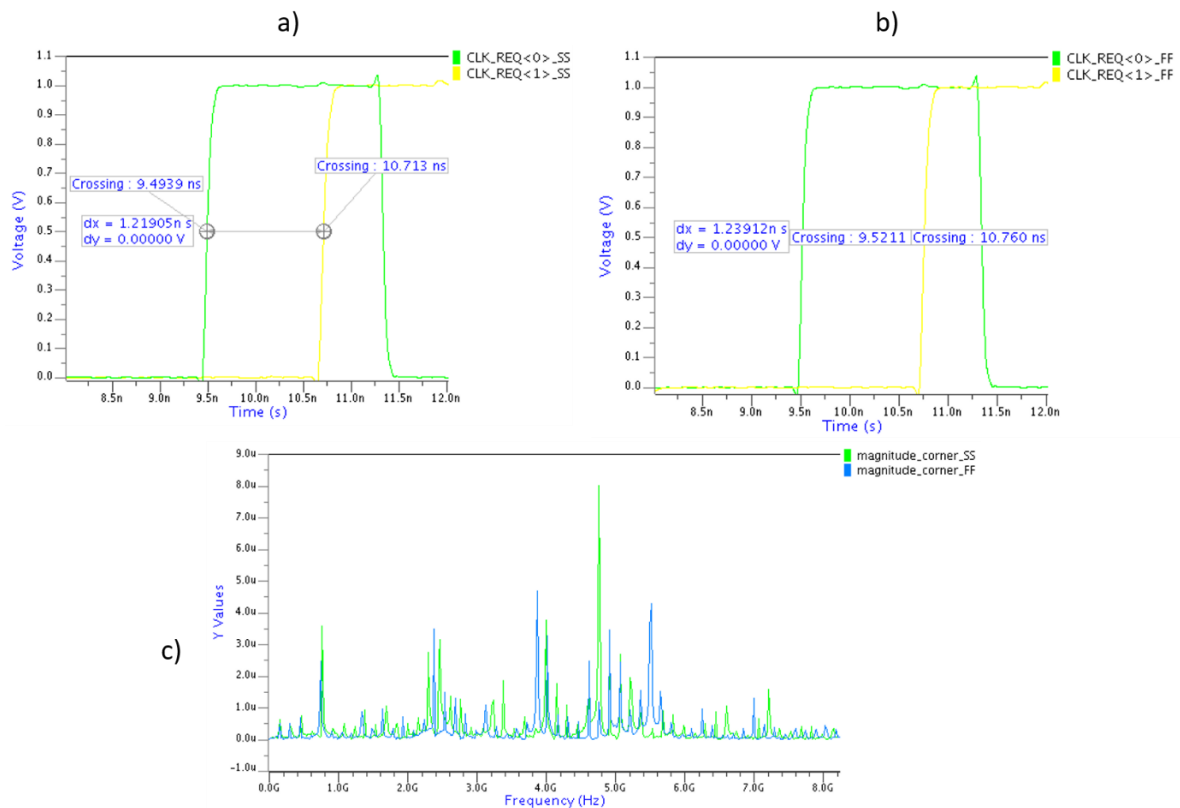


Figure V.29. Simulation avec l'extraction des parasites après le placement routage en CMOS 40 nm STMICROELECTRONICS de la première et deuxième requête pour a) des conditions pire cas b) des conditions meilleures cas avec un ajout de délais grâce aux PDL et c) comparaison des spectres électromagnétiques des deux conditions.

V. 6. Conclusion

La première partie de ce chapitre a présenté les paramètres que le concepteur peut changer dans son circuit pour atteindre ses spécifications électromagnétiques. Dans un premier temps, une conception asynchrone du circuit permet d'avoir des valeurs de délais différentes entre les étages. Cette variété apporte plus de possibilités à l'algorithme génétique pour créer des combinaisons et mettre en forme le rayonnement qu'un circuit désynchronisé (à partir d'une description synchrone). Ensuite, la résolution entre les valeurs possibles pour un délai est un critère très important. Les harmoniques présentes aux fréquences inversement proportionnelles à cette résolution ne pourront pas être réduites grâce aux délais. De plus, avec une résolution plus fine le nombre de délais possibles augmente ainsi que le nombre de combinaisons possibles. Enfin, le concepteur peut changer l'architecture de son circuit en divisant certains étages pour ajouter des paramètres à l'algorithme génétique et répartir plus facilement les appels de courant.

La deuxième partie présente notre premier circuit de test fabriqué en technologie CMOS 40 nm STMICROELECTRONICS. Nous avons conçu un circuit de cryptographie de norme de chiffrement avancé (*Advanced Encryption Standard, AES*) asynchrone sur 128 bits. Des délais

programmables ont été ajoutés entre les contrôleurs pour pouvoir mesurer différentes combinaisons de délais. La réalisation de ce circuit nous a permis de mettre en place le flot de conception présenté dans le chapitre IV.

La partie suivante montre les mesures que nous avons réalisées sur l'AES asynchrone. Dans un premier temps, nous avons validé le fonctionnement du circuit en réalisant des mesures temporelles sur le signal de test implémenté dans le circuit. Ce signal passe au niveau logique haut au début du calcul et au niveau bas à la fin. L'augmentation du temps de calcul avec les valeurs dans les délais programmables a été validée grâce à des mesures temporelles du signal de test. Ensuite, nous avons réalisé des mesures de courant. Elles nous permettent de voir que le courant change lorsque le circuit fonctionne mais pas de valider notre modèle de haut niveau de courant. Un régulateur à faible chute de tension et une capacité de 3 nF lissent le courant du circuit et ne nous permettent pas de visualiser correctement les pics de chaque étage. Enfin, des mesures électromagnétiques ont été réalisées avec différents instruments de mesures. Les mesures réalisées avec le scanner de surface ont permis de montrer l'efficacité de notre méthode de mise en forme du courant. Des combinaisons de délais trouvées à l'aide de la co-simulation pour réduire le rayonnement sur des plages de fréquences spécifiques ont été mesurées. Celles-ci ont prouvé que le spectre électromagnétique était déplacé en dehors de ces intervalles de fréquences.

Enfin, la dernière partie présente le deuxième circuit de test que nous avons conçu. Malheureusement, celui-ci n'a pas été fabriqué. L'objectif de ce circuit est de permettre de meilleures mesures de courant, de rayonnement et d'explorer différentes conditions de fonctionnement. L'architecture du circuit est plus simple, avec seulement cinq étages qui sont rebouclés, pour pouvoir corrélérer plus facilement avec les simulations. Il a été implémenté dans deux technologies différentes, suite à deux opportunités potentielles de fabrication, en BiCMOS 55 nm STMicroelectronics et en CMOS 40 nm STMicroelectronics. Ces deux circuits sont complètement finis et prêts à être envoyés si une nouvelle opportunité se présente. Nous avons également montré en simulations, que l'ajout de délais programmables peut permettre de palier aux variations du processus de fabrication, pour retrouver un rayonnement proche de celui désiré. Des études plus approfondies dans ce domaine seraient néanmoins intéressantes.

Conclusion Générale

Le fonctionnement des circuits synchrones, avec un signal d'horloge global, présente de sérieuses limitations quand on considère les nouveaux enjeux de conception, notamment avec des spécifications non fonctionnelles comme la consommation ou la compatibilité électromagnétique. Dans ce dernier cas, la capture simultanée des données dans tous les éléments de mémorisation crée d'important appels de courant sur l'alimentation, qui génèrent un fort rayonnement électromagnétique avec des harmoniques. Si ce rayonnement n'est pas maîtrisé, il crée une pollution électromagnétique qui peut venir perturber le fonctionnement d'autres circuits dans l'environnement, voire même s'auto-perturber. C'est pourquoi des normes de compatibilité électromagnétique ont été mises en place pour contrôler et contraindre les émissions des circuits en fonction de leurs applications.

Les circuits asynchrones présentent une alternative de conception très intéressante notamment pour les émissions électromagnétiques des circuits. En effet, leur fonctionnement exploite une synchronisation locale par « poignée de main » réalisée avec des contrôleurs distribués. Cette approche permet de répartir les appels de courant dans le temps ce qui se traduit par un étalement du spectre et donc une réduction d'amplitude des pics dans le spectre fréquentiel. C'est ainsi que des méthodes permettant de réduire qualitativement le spectre des circuits asynchrones ont été proposées. Il est à noter qu'il existe aussi des méthodes pour les circuits synchrones exploitant aussi une stratégie de modification des appels de courant comme, par exemple, l'introduction d'un jitter sur le signal d'horloge.

Dans notre méthode, nous avons également une approche quantitative et avons introduit un gabarit fréquentiel du rayonnement électromagnétique que le circuit devra respecter. Ainsi, dès l'étape de conception, le circuit pourra être défini par ses normes de compatibilité électromagnétique. Nous avons fait ici l'hypothèse que le spectre est fortement corrélé aux appels de courant sur l'alimentation. Notre solution a donc pour objectif de placer les appels de courant liés aux différents étages du circuit de manière à respecter le gabarit. Dans ce manuscrit, nous appliquons notre méthode à des circuits micropipelines. Néanmoins, elle pourrait être utilisée avec les autres classes de circuits asynchrones.

Pour obtenir la consommation de courant du circuit, nous avons développé un simulateur analogique rapide. Pour cela, le circuit est modélisé grâce à un réseau de Petri temporel, décrit en SystemVerilog. Les délais entre les contrôleurs sont insérés dans le modèle à l'aide d'un fichier standard de délais. Grâce à un fichier de stimuli, aussi en SystemVerilog, les temps des activations des étages sont extraits. Nous avons ensuite modélisé les pics de courant qui sont composés de deux exponentielles dos à dos. Les paramètres de chaque pic dépendent du temps de propagation du signal de requête, du nombre de registres et du chemin critique de la logique combinatoire de l'étage. Ainsi, la consommation totale du circuit est obtenue en associant à chaque instant de commutation le pic de courant correspondant. Cette

modélisation de haut niveau nous permet d'obtenir une estimation rapide des appels de courant du circuit en fonction des délais insérés dans le chemin de contrôle.

Ce simulateur a été intégré à notre flot de conception qui permet de contrôler le rayonnement électromagnétique. Toutes les étapes sont réalisées avec des outils standards utilisés dans l'industrie. La première étape correspond à décrire le circuit avec un langage de description du matériel au niveau registres (*Register Transfer Level, RTL*). Le gabarit fréquentiel à respecter est également défini. Ensuite, la synthèse logique est réalisée pour obtenir une description au niveau portes. Un flot d'analyse automatique du réseau de contrôle a été développé pour pouvoir définir les contraintes de la synthèse et de l'analyse temporelle statique, qui est l'étape suivante. Cette dernière permet d'extraire les valeurs de délais minimales à insérer dans le chemin de contrôle pour respecter les hypothèses temporelles et assurer le fonctionnement du circuit. Elle permet également d'obtenir les paramètres pour définir les pics de courant de chaque étage du circuit. Ensuite, une étape additionnelle a été ajoutée pour mettre en forme le spectre électromagnétique. Elle est composée d'un algorithme génétique, qui crée des combinaisons de délais pour le chemin de contrôle tandis que notre simulateur analogique rapide modélise la consommation de courant de chacune de ces combinaisons. Un environnement de co-simulation a été mis en place pour assurer les échanges d'informations entre l'algorithme et le simulateur. Une fois la courbe de courant obtenue, une transformée de Fourier est appliquée pour obtenir le spectre. Celui-ci est comparé avec le gabarit, défini au cours de la première étape. Tant que le spectre est au-dessus du gabarit, l'algorithme génère de nouvelles combinaisons. Une fois qu'une solution est trouvée, le circuit peut être implémenté avec cette combinaison de délais. Le placement et le routage du circuit sont réalisés de la même manière que dans un flot classique mais avec des contraintes spécifiques pour équilibrer le chemin de la requête et insérer les bons délais. Des scripts ont été développés pour automatiser les différentes étapes du flot. Le concepteur a en outre la possibilité de changer les paramètres du circuit à ce stade afin de trouver plus facilement une solution lors de la mise en forme du spectre.

Notre méthode a été utilisée pour concevoir deux circuits de test. Le premier a été fabriqué en technologie CMOS 40 nm STMicroelectronics, il correspond à un circuit de cryptographie de norme de chiffrement avancé (*Advanced Encryption Standard, AES*). Des délais programmables ont été insérés entre les contrôleurs dans le but de mesurer différentes combinaisons. Des mesures électromagnétiques, avec un scanner de surface relié à un analyseur de spectre, ont montré que des combinaisons de délais spécifiques permettent d'enlever le rayonnement sur des plages de fréquences spécifiques. Même si les émissions mesurées sont faibles, les mesures montrent que notre méthode fonctionne. Un deuxième circuit de test permettant des caractérisations plus fines a également été conçu. Malheureusement, les occasions pour le fabriquer ne se sont pas concrétisées. Néanmoins, il est prêt à être envoyé dans deux technologies différentes, en BiCMOS 55 nm STMicroelectronics et en CMOS 40 nm STMicroelectronics. Il nous a néanmoins permis de réaliser des simulations après placement et routage avec l'extraction des parasites et

l'insertion des capacités entre les fils. Nous avons ainsi démontré qu'en ajustant les délais après fabrication (lorsque le circuit est plus rapide que prévu par exemple) il est possible de se ramener à un spectre très proche de celui attendu.

Perspectives

Ce travail de thèse a donné des résultats très encourageants, notamment avec les mesures électromagnétiques du premier circuit de test. C'est pourquoi, il serait intéressant de continuer et d'approfondir ces recherches sur différents aspects.

La fabrication du deuxième circuit de test permettrait, du fait de sa structure plus simple, de mieux corréliser les mesures avec les simulations. Le circuit doit être intégré avec sa propre alimentation et sans régulateur à faible chute de tension pour faciliter les mesures de courant. Elles sont nécessaires pour valider ou améliorer notre modèle de pics de courant. La modularité de ce circuit permettrait aussi de tester beaucoup de configurations différentes, en faisant varier le nombre de registres qui commutent ou les délais sur le chemin de contrôle.

De plus une étude plus poussée sur l'impact des variations du processus de fabrication, des tensions et des températures (Process, Voltage, Temperature, PVT) est nécessaire. Des solutions, telle que l'insertion de délais programmables optimisés, pourraient permettre de compenser les variations de délais (liées aux procédés de fabrication) et de retrouver la combinaison désirée et, ainsi, le spectre correspondant. La réalisation du second circuit de test permettrait notamment de tester plusieurs pièces qui ne fonctionnent pas dans les mêmes conditions après fabrication et d'ajuster des délais pour trouver un rayonnement semblable.

Il serait aussi intéressant de développer un banc de mesures électromagnétiques spécifique. Il permettrait de mesurer le nouveau circuit de test, qui possède également des délais programmables, et aussi d'effectuer des mesures complémentaires sur le premier. En effet, le développement d'instruments de mesures spécifiques permettrait d'améliorer la qualité des mesures réalisées.

Références

- [1] C. M. Hung et K. Muhammad, « RF/Analog and Digital Faceoff-Friends or Enemies in an RF SoC », *VLSI Technology Systems and Applications (VLSI-TSA), 2010 International Symposium on*, p. 19-20, 2010.
- [2] R. Rossi, G. Torelli, et V. Liberali, « Model and verification of triple-well shielding on substrate noise in mixed-signal CMOS ICs », *ESSCIRC 2004 - 29th European Solid-State Circuits Conference (IEEE Cat. No.03EX705)*, p. 643-646, 2003.
- [3] C. Sitik, W. Liu, B. Taskin, et E. Salman, « Design Methodology for Voltage-Scaled Clock Distribution Networks », *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, n° 10, p. 3080-3093, oct. 2016.
- [4] H. Kawaguchi et T. Sakurai, « A reduced clock-swing flip-flop (RCSFF) for 63% power reduction », *IEEE Journal of Solid-State Circuits*, vol. 33, n° 5, p. 807-811, mai 1998.
- [5] S.H. Unger, *Asynchronous sequential switching circuits*. Wiley-Interscience, Chichester, Sussex, UK, 1969.
- [6] A. J. Martin, S. M. Burns, T. K. Lee, D. Borkovic, et P. J. Hazewindus, « The First Asynchronous Microprocessor: The Test Results », *SIGARCH Comput. Archit. News*, vol. 17, n° 4, p. 95–98, juin 1989.
- [7] S. M. Nowick et M. Singh, « Asynchronous Design—Part 1: Overview and Recent Advances », *IEEE Design Test*, vol. 32, n° 3, p. 5-18, juin 2015.
- [8] M. Renaudin, « Asynchronous circuits and systems: a promising design alternative », *Microelectronic Engineering*, vol. 54, n° 1-2, p. 133-149, 2000.
- [9] S. M. Nowick et M. Singh, « Asynchronous Design—Part 2: Systems and Methodologies », *IEEE Design Test*, vol. 32, n° 3, p. 19-28, juin 2015.
- [10] D.E. Muller, « Asynchronous logics and application to information processing », *Proc. Symp. on Application of Switching Theory in Space Technology*, p. 289-297, 1963.
- [11] M. Renaudin et J.B. Rigaud, « Etat de l'art sur la conception des circuits asynchrones : perspectives pour l'intégration des systèmes complexes », 2000.
- [12] I.E. Sutherland, « Micropipelines », *Communication of the ACM*, vol. 32, n° 6, p. 720-738, 1989.
- [13] A. Peeters et K. van Berkel, « Single-rail handshake circuits », in *Proceedings Second Working Conference on Asynchronous Design Methodologies*, 1995, p. 53-62.
- [14] J. Sparsø et S. Furber, *Principles of asynchronous circuit design: a systems perspective*. Boston: Kluwer, 2010.
- [15] S. Hauck, « Asynchronous design methodologies: an overview », *Proceedings of the IEEE*, vol. 83, n° 1, p. 69-93, 1995.
- [16] A. Kondratyev et K. Lwin, « Design of asynchronous circuits using synchronous CAD tools », *IEEE Design Test of Computers*, vol. 19, n° 4, p. 107-117, juill. 2002.
- [17] J.T. Udding, « A formal model for defining and classifying delay-insensitive circuits and systems », *Distributed Computing*, vol. 1, n° 4, p. 197-204, 1986.
- [18] A.J. Martin, « The limitations to delay-insensitivity in asynchronous circuits », présenté à Proc. 6th MIT Conf. on Advanced Research in VLSI, 1990, p. 263-278.
- [19] J.C. Ebergen, « A formal approach to designing delay-insensitive circuits », *Distributed Computing*, vol. 5, n° 3, p. 107-119, 1991.
- [20] A.J. Martin, « Compiling communicating processes into delay-insensitive VLSI circuits », *Distributed Computing*, vol. 1, n° 4, p. 226-234, 1986.

Références

- [21] R.E. Miller, *Switching Theory, Volume II: Sequential Circuits and Machines*. 1965.
- [22] P.A. Beerel, R.O. Ozdag, et M. Ferretti, *A designer's guide to asynchronous VLSI*. Cambridge ; New York: Cambridge University Press, 2010.
- [23] A. Charoy, *Parasites et perturbations des électroniques*. 1992.
- [24] T. S. Perry et L. Geppert, « Do portable electronics endanger flight? The evidence mounts », *IEEE Spectrum*, vol. 33, n° 9, p. 26-33, sept. 1996.
- [25] Institut National de Recherche et de Sécurité, « Champs électromagnétiques. Effets des champs électromagnétiques sur la santé - Risques - INRS », mai-2017. [En ligne]. Disponible sur: <http://www.inrs.fr/risques/champs-electromagnetiques/effets-sante.html>. [Consulté le: 16-avr-2019].
- [26] M. Mardiguian, *Manuel pratique de compatibilité électromagnétique*. 2003.
- [27] M. Ramdani *et al.*, « The Electromagnetic Compatibility of Integrated Circuits—Past, Present, and Future », *IEEE Transactions on Electromagnetic Compatibility*, vol. 51, n° 1, p. 78-100, 2009.
- [28] « International Electrotechnical Commission ». [En ligne]. Disponible sur: www.iec.ch. [Consulté le: 14-mars-2019]
- [29] Federal Comm. Commission, « In the Matter of Revision of the Commission's Rules Regarding Ultra-Wideband Transmission Systems », First Report and Order, ET Docket 98-153, FCC 02-8, 2002.
- [30] H. Chaudhary, N. Goyal, et N. Sah, « Dynamic Power Reduction Using Clock Gating: A Review », vol. 6, n° 1, p. 5, 2015.
- [31] D. Terrell et R.K. Keenan, *Digital Design for Interference Specifications: A Practical Handbook for EMI Suppression*. Newnes, 1997.
- [32] T. Steinecke, « Design-in for EMC on CMOS large-scale integrated circuits », présenté à IEEE EMC International Symposium. Symposium Record. International Symposium on Electromagnetic Compatibility, 2001, vol. 2, p. 910-915.
- [33] N. Srinivasan, N.S. Prakash, D. Shalakra, D. Sivaranjani, G.S. Sri Lakshmi, et B.B.T. Sundari, « Power Reduction by Clock Gating Technique », *Procedia Technology*, vol. 21, p. 631-635, janv. 2015.
- [34] S. Tamil Chindhu et N. Shanmugasundaram, « Clock Gating Techniques: An Overview », in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, 2018, p. 217-221.
- [35] J. Carlsson, K. Palmkvist, et L. Wanhammar, « A Clock Gating Circuit for Globally Asynchronous Locally Synchronous Systems », in *2006 NORCHIP*, 2006, p. 15-18.
- [36] D.K. Sharma, « Effects of Different Clock Gating Techniques on Design », vol. 3, n° 5, p. 4, 2012.
- [37] B. Pandey, J. Yadav, M. Pattanaik, et N. Rajoria, « Clock gating based energy efficient ALU design and implementation on FPGA », in *2013 International Conference on Energy Efficient Technologies for Sustainability*, 2013, p. 93-97.
- [38] K. B. Hardin, J. T. Fessler, et D. R. Bush, « Spread spectrum clock generation for the reduction of radiated emissions », présenté à Proceedings of IEEE Symposium on Electromagnetic Compatibility, 1994, p. 227-231.
- [39] H. Sadamura *et al.*, « Spread-spectrum clocking in switching regulators to reduce EMI », in *Proceedings. IEEE Asia-Pacific Conference on ASIC*, 2002, p. 141-144.
- [40] L. S. Nielsen et J. Sparso, « Designing asynchronous circuits for low power: an IFIR filter bank for a digital hearing aid », *Proceedings of the IEEE*, vol. 87, n° 2, p. 268-281, févr. 1999.
- [41] H. van Gageldonk, K. van Berkel, A. Peeters, D. Baumann, D. Gloor, et G. Stegmann, « An asynchronous low-power 80C51 microcontroller », in *Proceedings Fourth International Symposium on Advanced Research in Asynchronous Circuits and Systems*, 1998, p. 96-107.

- [42] S.B. Furber *et al.*, « AMULET2e: an asynchronous embedded controller », *Proceedings of the IEEE*, vol. 87, n° 2, p. 243-256, 1999.
- [43] G.F. Bouesse, N. Ninon, G. Sicard, M. Renaudin, A. Boyer, et E. Sicard, « Asynchronous logic VS Synchronous logic: Concrete results on electromagnetic emissions and conducted susceptibility », présenté à 6th International workshop on electromagnetic compatibility of integrated circuits, 2007, p. 99-102.
- [44] K. Van Berkel, R. Burgess, J. Kessels, M. Roncken, F. Schlij, et A. Peeters, « Asynchronous circuits for low power: a DCC error corrector », *IEEE Design & Test of Computers*, vol. 11, n° 2, p. 22-32, 1994.
- [45] N.C. Paver, P. Day, C. Farnsworth, D.L. Jackson, W.A. Lien, et J. Liu, « A Low-Power, Low Noise, Configurable Self-Timed DSP », *Advanced Research in Asynchronous Circuits and Systems, 1998. Proceedings. 1998 Fourth International Symposium on*, p. 32-42, 1998.
- [46] K. V. Berkel, M. B. Josephs, et S. M. Nowick, « Scanning the technology: applications of asynchronous circuits », *Proc. IEEE*, p. 223-233, 1999.
- [47] X. Fan *et al.*, « Frequency-Domain Optimization of Digital Switching Noise Based on Clock Scheduling », *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, n° 7, p. 982-993, juill. 2016.
- [48] M. Badaroglu, P. Wambacq, G. Van der Plas, S. Donnay, G.G.E. Gielen, et H.J. De Man, « Digital ground bounce reduction by supply current shaping and clock frequency Modulation », *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, n° 1, p. 65-76, 2005.
- [49] M. Krstic, X. Fan, M. Babic, E. Grass, T. Bjerregaard, et A. Yakovlev, « Reducing switching noise effects by advanced clock management », présenté à 11th International Workshop on the Electromagnetic Compatibility of Integrated Circuits (EMCCompo), 2017.
- [50] D. Panyasak, G. Sicard, et M. Renaudin, « A current shaping methodology for lowering EM disturbances in asynchronous circuits », *Microelectronics Journal*, vol. 35, n° 6, p. 531-540, 2004.
- [51] D. Panyasak, « Réduction de l'émission électromagnétique des circuits intégrés : l'alternative asynchrone », Institut National Polytechnique de Grenoble, 2004.
- [52] P.G. Paulin et J.P. Knight, « Force-directed scheduling for the behavioral synthesis of ASICs », *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 8, n° 6, p. 661-679, 1989.
- [53] M. Cazzaniga, P. Joubert Doriol, A. Sanna, E. Blanc, V. Liberali, et D. Pandini, « Evaluating the impact of substrate noise on conducted EMI in automotive microcontrollers », *Electromagnetic Compatibility of Integrated Circuits (EMC Compo), 2013 9th Intl Workshop on*, p. 129-133, 2013.
- [54] « European Telecommunications Standards Institute ». [En ligne]. Disponible sur: <https://www.etsi.org/>. [Consulté le: 09-avr-2019]
- [55] ISO/IEC 14443-2, « Identification cards — Contactless integrated circuit cards — Proximity cards — Part 2: Radio frequency power and signal interface », INTERNATIONAL STANDARD Third edition, juill. 2016.
- [56] ISO/IEC 14443-1, « Cards and security devices for personal identification — Contactless proximity objects — Part 1: Physical characteristics », INTERNATIONAL STANDARD Fourth edition, avr. 2018.
- [57] ISO/IEC 14443-3, « Cards and security devices for personal identification — Contactless proximity objects — Part 3: Initialization and anticollision », INTERNATIONAL STANDARD Fourth edition, juill. 2018.

Références

- [58] ISO/IEC 14443-4, « Cards and security devices for personal identification — Contactless proximity objects — Part 4: Transmission protocol », INTERNATIONAL STANDARD Fourth edition, juill. 2018.
- [59] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor: University of Michigan Press, 1975.
- [60] « Matlab Genetic Algorithm ». [En ligne]. Disponible sur: <https://fr.mathworks.com/help/gads/how-the-genetic-algorithm-works.html>. [Consulté le: 12-avr-2019]
- [61] C.A. Petri, « Communication With Automata », p. 97, 1966.
- [62] A.V. Yakovlev, A.M. Koelmans, A. Semenov, et D.J. Kinniment, « Modelling, analysis and synthesis of asynchronous control circuits using Petri nets », *Integration, the VLSI Journal*, vol. 21, n° 3, p. 143-170, 1996.
- [63] P. Merlin et D. Farber, « Recoverability of Communication Protocols--Implications of a Theoretical Study », *IEEE Transactions on Communications*, vol. 24, n° 9, p. 1036-1043, sept. 1976.
- [64] C. Ramchandani, « Analysis of asynchronous concurrent systems by Petri nets », Project MAC, TR-120, M.I.T., Cambridge, MA, 1974.
- [65] J. Simatic, A. Cherkaoui, F. Bertrand, R. Possamai Bastos, et L. Fesquet, « A Practical Framework for Specification, Verification, and Design of Self-Timed Pipelines », présenté à 23rd IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC), 2017, p. 65-72.
- [66] « IEEE Standard for SystemVerilog—Unified Hardware Design, Specification, and Verification Language », *IEEE Std 1800-2017 (Revision of IEEE Std 1800-2012)*, p. 1-1315, févr. 2018.
- [67] M. Singh et S.M. Nowick, « MOUSETRAP: High-Speed Transition-Signaling Asynchronous Pipelines », *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, n° 6, p. 684-698, 2007.
- [68] G. Gimenez, A. Cherkaoui, G. Cogniard, et L. Fesquet, « Static Timing Analysis of Asynchronous Bundled-Data Circuits », in *2018 24th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)*, 2018, p. 110-118.
- [69] J. Daemen et V. Rijmen, « AES Proposal: Rijndael », AES Algorithm Submission, 1999.
- [70] Department of Commerce NIST, « Announcing the Advanced Encryption Standard (AES) », 2001.

Publications de l'auteur

Conférences internationales

- S. Germain, S. Engels, et L. Fesquet, « A Design Flow for Shaping Electromagnetic Emissions in Micropipeline Circuits », in *24th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC 2018)*, Vienna, Austria, 2018.
- S. Germain, S. Engels, et L. Fesquet, « Event-Based Design Strategy for Circuit Electromagnetic Compatibility », in *3rd International Conference on Event-Based Control, Communication and Signal Processing (EBCCSP 2017)*, Funchal, Portugal, 2017, p. 1-7.
- S. Germain, S. Engels, et L. Fesquet, « Shaping Electromagnetic Emissions of Event-Driven Circuits Thanks to Genetic Algorithms », in *Third International Conference on Advances in Signal, Image, and Video Processing (SIGNAL 2018)*, Nice, France, 2018.
Best Paper Award

Conférences invitées

- L. Fesquet, S. Germain, J. Simatic, A. Cherkaoui, T. Le Pelleter, et S. Engels, « Event-based processing: a new paradigm for low-power », in *19th IEEE Mediterranean Electrotechnical Conference (IEEE Melecon'18)*, Marrakesh, Morocco, 2018.
- L. Fesquet, J. Simatic, A. Darwish, A. Cherkaoui, et S. Germain, « From events to data-driven processing », in *3rd International Conference on Event-Based Control, Communication and Signal Processing (EBCCSP 2017)*, Funchal, Portugal, 2017.

Article de revue

- S. Germain, S. Engels, et L. Fesquet, « A High Level Current Modeling for Shaping Electromagnetic Emissions in Micropipeline Circuits », *Journal of Low Power Electronics and Applications*, vol. 9, n° 1, 2019.

Autres

Un brevet est en cours de dépôt aux Etats-Unis.

Membre du comité de programme de la conférence SIGNAL 2019.

Titre : Contrôle du spectre électromagnétique d'un circuit numérique asynchrone

Résumé : La compatibilité électromagnétique des circuits est devenue un enjeu majeur en conception numérique. Des méthodes de conception existent déjà pour réduire de manière qualitative le rayonnement des circuits synchrones. Cependant, il est difficile de réduire les harmoniques du spectre dues au signal d'horloge. La logique asynchrone est une alternative très intéressante, qui permet d'éviter la commutation simultanée de tous les éléments de mémorisation et de supprimer les harmoniques de l'horloge. Nous avons développé un flot de conception pour mettre en forme le rayonnement électromagnétique et ainsi répondre aux normes de compatibilité électromagnétique. Notre méthode s'applique à des circuits asynchrones. Une analyse temporelle statique permet d'extraire les délais à insérer dans le chemin de contrôle nécessaire au respect des hypothèses temporelles. Cette analyse sert aussi à annoter le modèle du circuit dans un simulateur analogique rapide, qui a été développé spécifiquement pour extraire sa consommation de courant. Grâce à ce simulateur, en calculant la transformée de Fourier, il est possible de remonter au spectre électromagnétique. Un algorithme génétique est alors utilisé pour créer des combinaisons de délais, permettant de mettre en forme le spectre selon le gabarit spécifié. Des mesures électromagnétiques sur un circuit de test, fabriqué en technologie CMOS 40 nm STMicroelectronics, ont montré que notre méthode permet de contrôler le rayonnement électromagnétique du circuit.

Mots-clés : Circuits asynchrones, Compatibilité électromagnétique, Flot de conception

Title: Electromagnetic spectrum control of asynchronous digital circuits

Abstract: Electromagnetic Compatibility has become a major issue for the digital designers. Several techniques exist qualitatively reducing the electromagnetic emissions of synchronous circuits. Nevertheless, the clocked activity produces strong periodic current pulses on the power supply, generating harmonics in the electromagnetic spectrum. Contrarily, asynchronous designs, also known as clockless circuits, show a spread electromagnetic spectrum without harmonics. We have developed a design flow in order to shape the electromagnetic spectrum and meet electromagnetic compatibility standards. Our method only applies to asynchronous circuits. A static timing analysis is performed to extract the delays of the control path that guarantee the timing assumptions. This analysis is used to annotate the circuit model in a fast analog simulator, which was specifically developed to extract its current consumption. Thanks to this simulator and a Fast Fourier Transform, it is possible to get the electromagnetic spectrum. A genetic algorithm is then used to create delay combinations shaping the spectrum in order to match the targeted spectral mask. Electromagnetic measurements on a test chip, manufactured in STMicroelectronics CMOS 40nm technology, have shown that our method allows controlling the electromagnetic field of the circuit.

Keywords: Asynchronous circuits, Electromagnetic compatibility, Design flow

Thèse préparée au Laboratoire TIMA / Thesis prepared at TIMA Laboratory
Techniques de l'Informatique et de la Microélectronique pour l'Architecture des ordinateurs
Techniques of Informatics and Microelectronics for integrated systems Architecture
46 avenue Félix Viallet – 38031 GRENOBLE Cedex - France

ISBN: 978-2-11-129257-4